Cat. No. W394-IT2-07

Controllori programmabili

Serie SYSMAC CS
CS1G/H-CPU□□-EV1
CS1G/H-CPU□□H
CS1D-CPU□□H
CS1D-CPU□□S

Serie SYSMAC CJ
CJ1G-CPU□□
CJ1G/H-CPU□□H
CJ1M-CPU□□P
CJ1M-CPU□□

MANUALE DI PROGRAMMAZIONE

Informazioni generali in breve

1 Funzionamento della CPU
2 Programmazione
3 Funzioni di istruzione
4 Task

Advanced Industrial Automation

Serie SYSMAC CS

CS1G/H-CPU□□-EV1
CS1G/H-CPU□□H
CS1D-CPU□□H
CS1D-CPU□□S

Serie SYSMAC CJ

CJ1G-CPU□□
CJ1G/H-CPU□□H
CJ1G-CPU□□ P
CJ1M-CPU□□

Controllori programmabili

Manuale di programmazione

Revisione: luglio 2004

Avviso

I prodotti OMRON sono destinati all'uso da parte di un operatore qualificato secondo le procedure appropriate e solo per gli scopi descritti in questo manuale.

Nel presente manuale le precauzioni sono indicate e classificate in base alle convenzioni riportate di seguito Attenersi sempre alle istruzioni fornite. La mancata osservanza di tali precauzioni potrebbe causare lesioni a persone o danni a proprietà.

⚠ PERICOLO Indica una situazione di immediato pericolo che, se non evitata, sarà causa di lesioni gravi o mortali.

⚠ AVVERTENZA Indica una situazione di potenziale pericolo che, se non evitata, può essere causa di lesioni gravi o mortali.

⚠ Attenzione Indica una situazione di potenziale pericolo che, se non evitata, può essere causa di lesioni non gravi a persone o danni alla proprietà.

Riferimenti ai prodotti OMRON

Tutti i nomi di prodotti OMRON contenuti nel presente manuale iniziano con lettera maiuscola. Anche per la parola "Modulo" viene utilizzata l'iniziale maiuscola quando si riferisce a un prodotto OMRON, indipendentemente dal fatto che faccia o meno parte del nome proprio del prodotto.

L'abbreviazione "Ch", che compare su alcuni display e prodotti OMRON, spesso corrisponde a "canale", termine che viene in alcuni casi abbreviato come "Cnl" nella documentazione.

L'abbreviazione "PLC" indica un controllore programmabile. È tuttavia possibile che in alcuni dispositivi di programmazione venga visualizzata l'abbreviazione "PC" ad indicare il controllore programmabile.

Indicazioni visive

Nella colonna sinistra del manuale sono riportate le seguenti intestazioni per facilitare l'individuazione dei diversi tipi di informazioni.

Nota Indica informazioni di particolare rilevanza per un efficiente e vantaggioso utilizzo del prodotto.

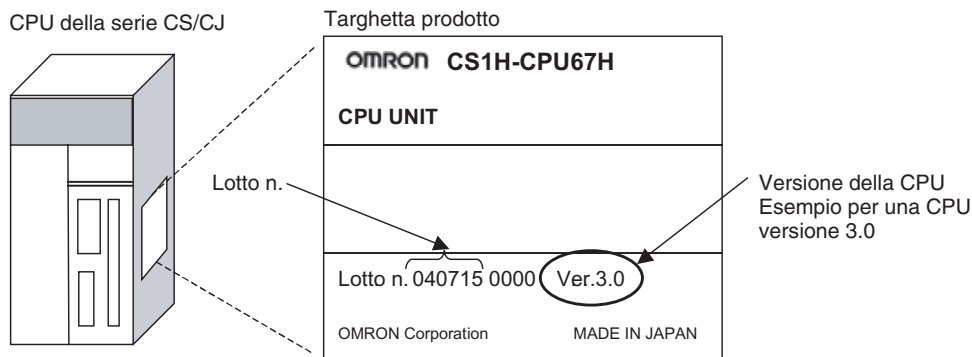
1,2,3... 1. Indica un qualche tipo di elenco, quali procedure, elenchi di controllo, ecc.

Versioni delle CPU della serie CS/CJ

Versioni delle CPU

Per gestire le CPU della serie CS/CJ in base alle differenze di funzionalità presenti negli aggiornamenti, è stato introdotto il concetto di "versione della CPU". Questo concetto si applica alle CPU CS1-H, CJ1-H, CJ1M e CS1D.

Notazione delle versioni delle CPU sui prodotti

La versione della CPU è riportata a destra del numero di lotto, sulla targhetta dei prodotti per i quali vengono gestite le versioni, come illustrato di seguito.

- Le CPU CS1-H, CJ1-H e CJ1M (eccetto i modelli di base) prodotti entro il 4 novembre del 2003 non riportano alcun numero di versione, ovvero la posizione della versione della CPU sopra illustrata risulta vuota.
- La versione delle CPU CS1-H, CJ1-H e CJ1M nonché delle CPU CS1D per sistemi a singola CPU inizia dalla versione 2.0.
- La versione delle CPU CS1D per sistemi a due CPU inizia dalla versione 1.1.
- Le CPU per le quali non viene fornita una versione sono definite *CPU precedenti alla versione □.□, ad esempio CPU precedenti alla versione 2.0* e *CPU precedenti alla versione 1.1.*

Verifica delle versioni delle CPU tramite software di supporto

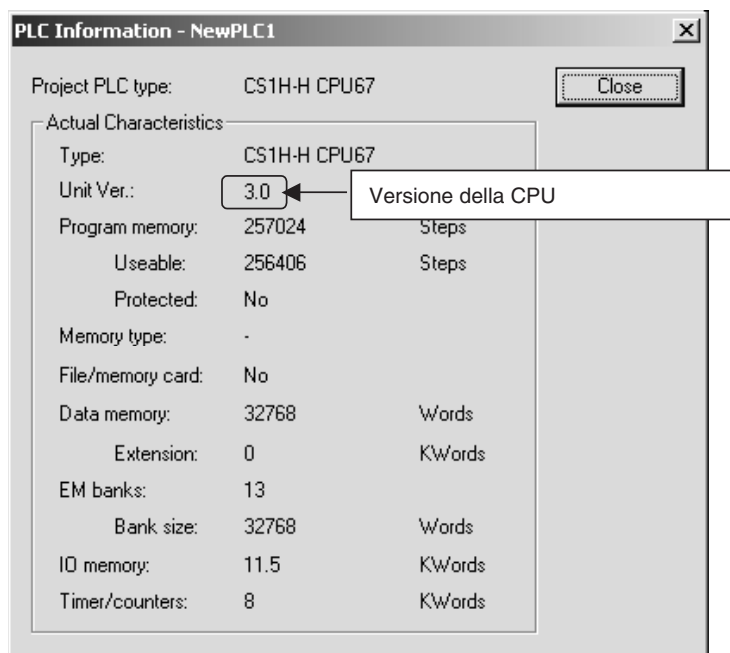
È possibile verificare la versione della CPU tramite CX-Programmer versione 4.0 utilizzando uno dei due metodi riportati di seguito.

- Utilizzo della finestra ***PLC Information*** (Informazioni sul PLC)
- Utilizzo della finestra ***Unit Manufacturing Information*** (Informazioni sulla produzione della CPU). Questo metodo può essere utilizzato anche per i Moduli di I/O speciale e i Moduli CPU bus.

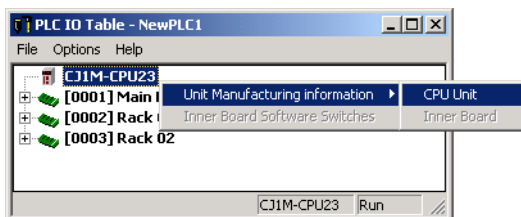
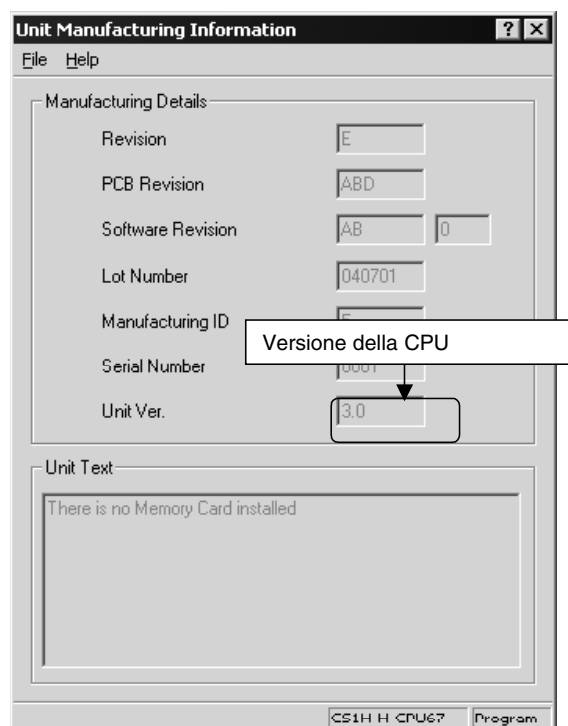
Nota Non è possibile verificare le versioni utilizzando CX-Programmer versione 3.3 o precedenti.

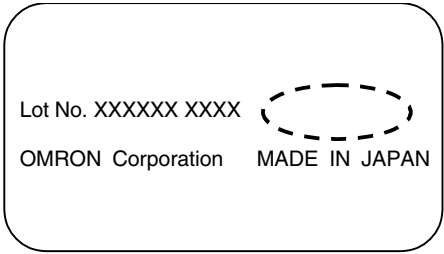
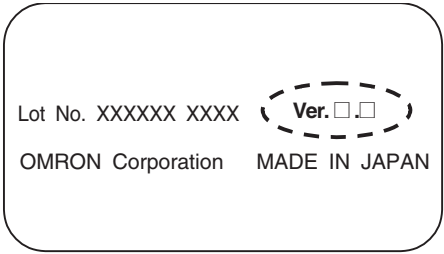
Informazioni sul PLC

- Se si conoscono il tipo di dispositivo e il tipo di CPU, selezionarli nella finestra di dialogo *Change PLC* (Cambia PLC), connettersi in linea e selezionare ***PLC - Edit - Information*** (PLC - Modifica - Informazioni) dai menu.
- Se non si conoscono il tipo di dispositivo e il tipo di CPU, ma si è connessi direttamente alla CPU tramite una linea seriale, selezionare ***PLC - Auto Online*** (PLC - Connessione in linea automatica) per stabilire una connessione in linea, quindi selezionare ***PLC - Edit - Information*** (PLC - Modifica - Informazioni).

In entrambi i casi, verrà visualizzata la finestra di dialogo *PLC Information* (Informazioni sul PLC) riportata di seguito.

Utilizzare la finestra sopra illustrata per verificare la versione della CPU.

Informazioni sulla produzione della CPU

Nella finestra *I/O Table* (Tabella di I/O) fare clic con il pulsante destro del mouse e selezionare ***Unit Manufacturing information - CPU Unit.*** (Informazioni sulla produzione del modulo - CPU).

Verrà visualizzata la finestra di dialogo *Unit Manufacturing information* (Informazioni sulla produzione della CPU) riportata di seguito.

Utilizzare la finestra sopra illustrata per verificare la versione della CPU connessa in linea.

Utilizzo delle etichette delle versioni delle CPU

Con la CPU vengono fornite le etichette delle versioni riportate di seguito.

È possibile apporre queste etichette sulla parte frontale delle CPU precedenti per distinguerle da quelle di versioni differenti.

Notazione delle versioni delle CPU

Nel presente manuale la versione di una CPU viene fornita come illustrato nella seguente tabella.

Targhetta prodotto / Descrizione	CPU su cui non viene fornita alcuna versione	CPU per le quali viene fornita una versione (Ver. □.□)
	Lot No. XXXXXX XXXX OMRON Corporation MADE IN JAPAN	Lot No. XXXXXX XXXX Ver. □.□ OMRON Corporation MADE IN JAPAN
Designazione di singole CPU (ad esempio CS1H-CPU67H)	CPU CS1-H precedenti alla versione 2.0	CPU CS1H-CPU67H versione □.□
Designazione dei gruppi delle CPU (ad esempio le CPU CS1-H)	CPU CS1-H precedenti alla versione 2.0	CPU CS1-H versione □.□
Designazione di un'intera serie di CPU (ad esempio le CPU della serie CS)	CPU della serie CS precedenti alla versione 2.0	CPU della serie CS versione □.□

Versioni di CPU e numeri di lotto

Serie	Modello			Dati di produzione						
				Precedente	Settembre 2003	Ottobre 2003	Novembre 2003	Dicembre 2003	Giugno 2004	Successiva
Serie CS	CPU CS1		CS1□-CPU□□	Nessuna versione della CPU						
	CPU CS1-V1		CS1□-CPU□□-V1	Nessuna versione della CPU						
	CPU CS1-H		CS1□-CPU□□H	CPU precedenti alla versione 2.0			CPU versione 2.0 (N. lotto da 031105 in poi)			CPU versione 3.0 (N. lotto da 040622 in poi)
	CPU CS1D	CPU per sistemi a due CPU	CS1D-CPU□□H	CPU precedenti alla versione 1.1			CPU versione 1.1 (N. lotto da 031120 in poi)			
		CPU per sistemi a singola CPU	CS1D-CPU□□S					CPU versione 2.0 (N. lotto da 031215 in poi)		
Serie CJ	CPU CJ1		CJ1G-CPU□□	CPU precedenti alla versione 2.0						
	CPU CJ1-H		CJ1□-CPU□□H	CPU precedenti alla versione 2.0			CPU versione 2.0 (N. lotto da 031105 in poi)			CPU versione 3.0 (N. lotto da 040623 in poi)
	CPU CJ1M eccetto modelli di base		CJ1M-CPU□□	CPU precedenti alla versione 2.0			CPU versione 2.0 (N. lotto da 031105 in poi)			CPU versione 3.0 (N. lotto da 040624 in poi)
	CPU CJ1M, modelli di base		CJ1M-CPU11/21			CPU versione 2.0 (N. lotto da 031002 in poi)				CPU versione 3.0 (N. lotto da 040629 in poi)
Software di supporto	CX-Programmer		WS02-CXPC1-EV□	Versione 3.2		Versione 3.3		Versione 4.0		Versione 5.0

Funzioni supportate dalla versione della CPU

CPU CS1-H (CS1□-CPU□□H)

Funzione		Versione della CPU	
		CPU precedenti alla versione 2.0	**CPU versione 2.0**
Scaricamento e caricamento di singoli task		---	OK
Protezione da lettura migliorata mediante password		---	OK
Protezione da scrittura dai comandi FINS inviati alle CPU tramite reti		---	OK
Connessioni di rete in linea senza tabelle degli I/O		---	OK
Comunicazioni tramite un massimo di 8 livelli di rete		---	OK
Connessione in linea ai PLC tramite PT della serie NS		OK a partire dal numero di lotto 030201	OK
Impostazione dei canali del primo slot		OK per un massimo di 8 gruppi	OK per un massimo di 64 gruppi
Trasferimento automatico all'accensione del sistema senza un file dei parametri		---	OK
Rilevamento automatico del metodo di assegnazione degli I/O per il trasferimento automatico all'accensione del sistema		---	---
Ora di inizio/fine del funzionamento		---	OK
Nuove istruzioni di applicazione	MILH, MILR, MILC	---	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	OK
	BCMP2	---	OK
	GRY	OK a partire dal numero di lotto 030201	OK
	TPO	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	OK
	Lettura/Scrittura da/su Moduli CPU bus tramite IORD/IOWR	OK a partire dal numero di lotto 030418	OK
	PRV2	---	---

CPU CS1D

Funzione		CPU CS1D per sistemi a due CPU (CS1D-CPU□□H)		CPU CS1D per sistemi a singola CPU (CS1D-CPU□□S)
		CPU precedenti alla versione 1.1	CPU versione 1.1	CPU versione 2.0
Funzione esclusiva delle CPU CS1D	A due CPU	OK	OK	---
	Sostituzione in linea	OK	OK	OK
	Due Moduli di alimentazione	OK	OK	OK
	Due Moduli Controller Link	OK	OK	OK
	Due Moduli Ethernet	---	OK	OK
Scaricamento e caricamento di singoli task		---	---	OK
Protezione da lettura migliorata mediante password		---	---	OK
Protezione da scrittura dai comandi FINS inviati alle CPU tramite reti		---	---	OK
Connessioni di rete in linea senza tabelle degli I/O		---	---	OK
Comunicazioni tramite un massimo di 8 livelli di rete		---	---	OK
Connessione in linea ai PLC tramite PT della serie NS		---	---	OK
Impostazione dei canali del primo slot		---	---	OK per un massimo di 64 gruppi
Trasferimento automatico all'accensione del sistema senza un file dei parametri		---	---	OK
Rilevamento automatico del metodo di assegnazione degli I/O per il trasferimento automatico all'accensione del sistema		---	---	---
Ora di inizio/fine del funzionamento		---	OK	OK
Nuove istruzioni di applicazione	MILH, MILR, MILC	---	---	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	---	OK
	BCMP2	---	---	OK
	GRY	---	---	OK
	TPO	---	---	OK
	DSW, TKY, HKY, MTR, 7SEG	---	---	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	---	OK
	Lettura/Scrittura da/su Moduli CPU bus tramite IORD/IOWR	---	---	OK
	PRV2	---	---	---

CPU CJ1-H/CJ1M

Funzione		CPU CJ1-H (CJ1□-CPU□□H)		CPU CJ1M, eccetto modelli di base (CJ1M-CPU□□)		CPU CJ1M, modelli di base (CJ1M-CPU11/21)
		CPU precedenti alla versione 2.0	CPU versione 2.0	CPU precedenti alla versione 2.0	CPU versione 2.0	CPU versione 2.0
Scaricamento e caricamento di singoli task		---	OK	---	OK	OK
Protezione da lettura migliorata mediante password		---	OK	---	OK	OK
Protezione da scrittura dai comandi FINS inviati alle CPU tramite reti		---	OK	---	OK	OK
Connessioni di rete in linea senza tabelle degli I/O		OK, ma solo se all'accensione del sistema è impostata l'assegnazione delle tabelle degli I/O	OK	OK, ma solo se all'accensione del sistema è impostata l'assegnazione delle tabelle degli I/O	OK	OK
Comunicazioni tramite un massimo di 8 livelli di rete		OK per un massimo di 8 gruppi	OK per un massimo di 64 gruppi	OK per un massimo di 8 gruppi	OK per un massimo di 64 gruppi	OK per un massimo di 64 gruppi
Connessione in linea ai PLC tramite PT della serie NS		OK a partire dal numero di lotto 030201	OK	OK a partire dal numero di lotto 030201	OK	OK
Impostazione dei canali del primo slot		---	OK	---	OK	OK
Trasferimento automatico all'accensione del sistema senza un file dei parametri		---	OK	---	OK	OK
Rilevamento automatico del metodo di assegnazione degli I/O per il trasferimento automatico all'accensione del sistema		---	OK	---	OK	OK
Ora di inizio/fine del funzionamento		---	OK	---	OK	OK
Nuove istruzioni di applicazione	MILH, MILR, MILC	---	OK	---	OK	OK
	=DT, <>DT, <DT, <=DT, >DT, >=DT	---	OK	---	OK	OK
	BCMP2	---	OK	OK	OK	OK
	GRY	OK a partire dal numero di lotto 030201	OK	OK a partire dal numero di lotto 030201	OK	OK
	TPO	---	OK	---	OK	OK
	DSW, TKY, HKY, MTR, 7SEG	---	OK	---	OK	OK
	EXPLT, EGATR, ESATR, ECHRD, ECHWR	---	OK	---	OK	OK
	Lettura/Scrittura da/su Moduli CPU bus tramite IORD/IOWR	---	OK	---	OK	OK
	PRV2	---	---	---	OK, ma solo per i modelli con I/O integrati	OK, ma solo per i modelli con I/O integrati

Funzioni supportate dalle CPU versione 3.0 o successiva

CPU CS1-H (CS1□-CPU□□H)

Funzione		Versione della CPU	
		CPU precedenti alla versione 2.0	Versione 3.0
Blocchi funzione (supportati da CX-Programmer versione 5.0 o successiva)		---	OK
Gateway seriale (conversione dei comandi FINS in comandi CompoWay/F a livello di porta seriale integrata)		---	OK
Memoria dei commenti (nella memoria flash interna)		---	OK
Dati di backup semplice espansi		---	OK
Nuove istruzioni di applicazione	TXDU(256), RXDU(255) (supporto per la comunicazione senza protocollo tra Moduli di comunicazione seriale e CPU della versione 1.2 o successiva)	---	OK
	Istruzioni di conversione del modello: XFERC(565), DISTC(566), COLLC(567), MOVBC(568), BCNTC(621)	---	OK
	Istruzioni speciali per blocchi funzione: GETID(286)	---	OK
Funzioni di istruzione addizionali	Istruzioni TXD(235) e RXD(236) (supporto per la comunicazione senza protocollo tra Schede di comunicazione seriale e CPU della versione 1.2 o successiva)	---	OK

CPU CS1D Le CPU versione 3.0 non sono supportate.

CPU CJ1-H/CJ1M (CJ1□-CPU□□H, CJ1G-CPU□□P, CJ1M-CPU□□)

Funzione		Versione della CPU	
		CPU precedenti alla versione 2.0	Versione 3.0
Blocchi funzione (supportati da CX-Programmer versione 5.0 o successiva)		---	OK
Gateway seriale (conversione dei comandi FINS in comandi CompoWay/F a livello di porta seriale integrata)		---	OK
Memoria dei commenti (nella memoria flash interna)		---	OK
Dati di backup semplice espansi		---	OK
Nuove istruzioni di applicazione	TXDU(256), RXDU(255) (supporto per la comunicazione senza protocollo tra Moduli di comunicazione seriale e CPU della versione 1.2 o successiva)	---	OK
	Istruzioni di conversione del modello: XFERC(565), DISTC(566), COLLC(567), MOVBC(568), BCNTC(621)	---	OK
	Istruzioni speciali per blocchi funzione: GETID(286)	---	OK
Funzioni di istruzione addizionali	Istruzioni PRV(881) e PRV2(883): aggiunti metodi ad alta frequenza per il calcolo della frequenza di impulsi (solo CPU CJ1M).	---	OK

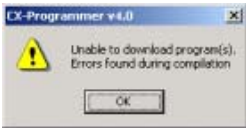
Versioni delle CPU e dispositivi di programmazione

Per consentire l'utilizzo delle funzioni aggiunte alla CPU versione 2.0, è necessario utilizzare CX-Programmer versione 4.0 o successiva.

Per consentire l'utilizzodei blocchi funzione aggiunti alle CPU versione 3.0, è necessario utilizzare CX-Programmer versione 5.0 o successiva.

Nelle seguenti tabelle è illustrata la relazione tra le versioni delle CPU e le versioni di CX-Programmer.

Versioni delle CPU e dispositivi di programmazione

CPU	Funzioni		CX-Programmer				Console di programmazione
			Versione 3.2 o precedente	Versione 3.3	Versione 4.0	Versione 5.0 o successiva	
CPU CJ1M, modelli di base, versione della CPU 2.0	Funzioni aggiuntive della versione 2.0	Con utilizzo delle nuove funzioni	---	---	OK	OK	Nessuna restrizione
		Senza utilizzo delle nuove funzioni	---	OK	OK	OK	
CPU CS1-H, CJ1-H e CJ1M, eccetto modelli di base, versione della CPU 2.0	Funzioni aggiuntive della versione 2.0	Con utilizzo delle nuove funzioni	---	---	OK	OK	
		Senza utilizzo delle nuove funzioni	OK	OK	OK	OK	
CPU CS1D per sistemi a singola CPU, versione della CPU 2.0	Funzioni aggiuntive della versione 2.0	Con utilizzo delle nuove funzioni	---	---	OK	OK	
		Senza utilizzo delle nuove funzioni				OK	
CPU CS1D per sistemi a due CPU, versione della CPU 1.	Funzioni aggiuntive della versione 1.1	Con utilizzo delle nuove funzioni	---	---	OK	OK	
		Senza utilizzo delle nuove funzioni	OK	OK	OK	OK	
CPU della serie CS/CJ versione 3.0	Blocchi funzione aggiunti per la versione 3.0	Con utilizzo dei blocchi funzione	---	---	---	OK	
		Senza utilizzo dei blocchi funzione	OK	OK	OK	OK	

Nota Come illustrato sopra, se non si utilizzano le funzioni aggiunte alla versione 2.0 o alla versione 1.1, non è necessario eseguire l'aggiornamento a CX-Programmer versione 4.0.

Impostazione del tipo di dispositivo

La versione della CPU non influisce sull'impostazione effettuata per il tipo di dispositivo in CX-Programmer. Selezionare il tipo di dispositivo come illustrato nella seguente tabella, indipendentemente dalla versione della CPU.

Serie	Gruppo CPU	Modello CPU	Impostazione del tipo di dispositivo in CX-Programmer versione 4.0 o successiva
Serie CS	CPU CS1-H	CS1G-CPU□□H	CS1G-H
		CS1H-CPU□□H	CS1H-H
	CPU CS1D per sistemi a due CPU	CS1D-CPU□□H	CS1D-H (o CS1H-H)
	CPU CS1D per sistemi a singola CPU	CS1D-CPU□□S	CS1D-S
Serie CJ	CPU CJ1-H	CJ1G-CPU□□H	CJ1G-H
		CJ1H-CPU□□H	CJ1H-H
	CPU CJ1M	CJ1M-CPU□□	CJ1M

Risoluzione dei problemi relativi alle versioni delle CPU in CX-Programmer

Problema	Causa	Soluzione
CX-Programmer v4.0 Unable to download program(s). Errors found during compilation OK Dopo la visualizzazione del messaggio sopra riportato, verrà visualizzato un errore di compilazione nella scheda *Compile* (Compila) della finestra di output.	Utilizzando CX-Programmer versione 4.0 o successiva, si è tentato di scaricare un programma che contiene istruzioni supportate solo dalle CPU versione 2.0 o successiva in una CPU precedente alla versione 2.0.	Controllare il programma o cambiare la CPU in cui si desidera scaricare il programma con una CPU versione 2.0 o successiva.
PLC Setup Error Unable to transfer the settings since they include setting items which are not supported by the connecting target CPU unit Check the version of the target CPU unit or the following PLC Settings, and transfer the settings again. - FINS Protection Settings for FINS write protection via network OK	Utilizzando CX-Programmer versione 4.0 o successiva, si è tentato di scaricare una configurazione del PLC che contiene impostazioni supportate solo dalle CPU versione 2.0 o successiva, ovvero non impostate sui valori predefiniti, in una CPU precedente alla versione 2.0.	Controllare le impostazioni della configurazione del PLC o cambiare la CPU in cui si desidera scaricare le impostazioni con una CPU versione 2.0 o successiva.
"????" viene visualizzato in un programma trasferito dal PLC a CX-Programmer.	Si è utilizzato CX-Programmer versione 3.3 o precedente per caricare un programma che contiene istruzioni supportate solo dalle CPU versione 2.0 o successiva da una CPU versione 2.0 o successiva.	Non è possibile caricare le nuove istruzioni utilizzando CX-Programmer versione 3.3 o precedente. Utilizzare CX-Programmer versione 4.0 o successiva.

INDICE

INDICE

Informazioni sul manuale

Il presente manuale fornisce informazioni sulla programmazione delle CPU per il controllori programmabili (PLC) della serie CS/CJ e include i capitoli descritti nella pagina successiva. I prodotti della serie CS e della serie CJ sono suddivisi come illustrato nella seguente tabella.

Modulo	Serie CS	Serie CJ
CPU	CPU CS1-H: CS1H-CPU□□H CS1G-CPU□□H	CPU CJ1-H: CJ1H-CPU□□H CJ1G-CPU□□H
	CPU CS1: CS1H-CPU□□-EV1 CS1G-CPU□□-EV1	CPU CJ1: CJ1G-CPU□□-EV1 CPU CJ1M: CJ1M-CPU□□
	CPU CS1D: CPU CS1D per sistemi a due CPU: CS1D-CPU□□H CPU CS1D per sistemi a singola CPU: CS1D-CPU□□S CPU con processo CS1D: CS1D-CPU□□P	
Moduli di I/O di base	Moduli di I/O di base della serie CS	Moduli di I/O di base della serie CJ
Moduli di I/O speciale	Moduli di I/O speciale della serie CS	Moduli di I/O speciale della serie CJ
Moduli CPU bus	Moduli CPU bus della serie CS	Moduli CPU bus della serie CJ
Moduli di alimentazione	Moduli di alimentazione della serie CS	Moduli di alimentazione della serie CJ

Leggere attentamente il presente manuale e tutti i manuali correlati elencati nella tabella riportata nella pagina seguente e accertarsi di avere ben compreso le informazioni fornite prima di procedere all'utilizzo o all'installazione delle CPU della serie CS/CJ in un sistema PLC.

Il manuale contiene i seguenti capitoli:

Capitolo 1 descrive la struttura di base e il funzionamento della CPU.

Capitolo 2 fornisce informazioni di base sulla creazione, la verifica e l'immissione di programmi.

Capitolo 3 fornisce una descrizione generale delle istruzioni che è possibile utilizzare per la creazione di programmi utente.

Capitolo 4 descrive il funzionamento dei task.

Capitolo 5 descrive le funzioni utilizzate per la gestione della memoria file.

Capitolo 6 fornisce informazioni dettagliate sulle funzioni avanzate, quali tempo di ciclo ed elaborazione veloce, registri indice, funzioni per la comunicazione seriale, di avvio e manutenzione, di diagnostica e debug, funzioni dei dispositivi di programmazione e impostazioni dei tempi di risposta di ingresso per i Moduli di I/O di base CJ.

Capitolo 7 descrive i processi utilizzati per il trasferimento del programma alla CPU e le funzioni per la verifica e il debug del programma.

Nelle ***Appendici*** viene effettuato il confronto tra i Moduli delle serie CS e della serie CJ e vengono inoltre fornite informazioni relative alle limitazioni di utilizzo dei Moduli di I/O speciale C200H e alle modifiche apportate ai sistemi Host Link.

Informazioni sul manuale (continua)

Nome	N. cat.	Descrizione
Manuale di riferimento delle istruzioni dei controllori programmabili per la serie SYSMAC CS/CJ CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1G/H-CPU□□H, CJ1M-CPU□□	W394	Fornisce informazioni sulla programmazione e altri metodi per l'utilizzo delle funzioni dei PLC della serie CS/CJ (il presente manuale).
Manuale dell'operatore dei controllori programmabili per la serie SYSMAC CS CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H	W339	Fornisce informazioni generali sui PLC della serie CS, ne descrive le caratteristiche di progettazione, le procedure di installazione e manutenzione nonché altre operazioni di base per l'utilizzo dei PLC.
Manuale dell'operatore dei controllori programmabili per la serie SYSMAC CJ CJ1G/H-CPU□□H, CJ1M-CPU□□, CJ1M-CPU□□ CJ1G-CPU□□	W393	Fornisce informazioni generali sui PLC della serie CJ, ne descrive le caratteristiche di progettazione, le procedure di installazione e manutenzione nonché altre operazioni di base per l'utilizzo dei PLC
Manuale dell'operatore degli I/O integrati per la serie SYSMAC CJ CJ1M-CPU21/22/23	W395	Descrive le funzioni degli I/O integrati delle CPU CJ1M
SYSMAC CS Series CS1D-CPU□□H CPU Units CS1D-CPU□□S CPU Units CS1D-DPL01 Duplex Unit CS1D-PA207R Power Supply Unit Duplex System Operation Manual	W405	Fornisce informazioni generali e descrive le caratteristiche di progettazione, le procedure di installazione e manutenzione nonché altre operazioni di base per un sistema duplex basato sulla CPU CS1D.
SYSMAC CS/CJ Series CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1M-CPU□□, CJ1G-CPU□□P, CJ1G/H-CPU□□H Programmable Controllers Instructions Reference Manual	W340	Fornisce una descrizione delle istruzioni di programmazione utilizzate nei diagrammi ladder supportate dai PLC della serie CS/CJ.
SYSMAC CS/CJ Series CQM1H-PRO01-E, C200H-PRO27-E, CQM1-PRO01-E Programming Consoles Operation Manual	W341	Fornisce informazioni sui metodi di programmazione e utilizzo dei PLC della serie CS/CJ mediante una Console di programmazione.
SYSMAC CS/CJ Series CS1G/H-CPU□□-EV1, CS1G/H-CPU□□H, CS1D-CPU□□H, CS1D-CPU□□S, CJ1G-CPU□□, CJ1G/H-CPU□□H, CJ1G-CPU□□P, CJ1M-CPU□□, CS1W-SCB21-V1/41-V1, CS1W-SCU21-V1, CJ1W-SCU21-V1/41-V1 Communications Commands Reference Manual	W342	Descrive i comandi di comunicazione FINS e dei PLC della serie C (Host Link) utilizzati con i PLC della serie CS/CJ.
SYSMAC WS02-CXP□□-E CX-Programmer Operation Manual Version 3.□	W414	Fornisce informazioni sull'utilizzo di CX-Programmer, un dispositivo di programmazione software che supporta i PLC della serie CS/CJ, e del programma CX-Net incluso in CX-Programmer.
SYSMAC WS02-CXP□□-E CX-Programmer Operation Manual Version 4.□	W425	
SYSMAC WS02-CXP□□-E CX-Programmer Operation Manual Version 5.□	W437	
SYSMAC WS02-CXP□□-E CX-Programmer Operation Manual Function Blocks	W438	Descrive specifiche e metodi operativi relativi ai blocchi funzione. Queste informazioni sono rilevanti solo se si utilizzano i blocchi funzione con CX-Programmer versione 5.0 e CPU CJ1-H/CJ1M versione 3.0. Per informazioni su altre funzionalità di CX-Programmer versione 5.0, fare riferimento al manuale *CX-Programmer Operation Manual Version 5.□* (W437*)*. 5.0.

Nome	N. cat.	Descrizione
SYSMAC CS/CJ Series CS1W-SCB21-V1/41-V1, CS1W-SCU21-V1, CJ1W-SCU21-V1/41-V1 Serial Communications Boards/Units Operation Manual	W336	Descrive l'utilizzo dei Moduli e delle Schede di comunicazione seriale per la comunicazione seriale con i dispositivi esterni, incluso l'impiego dei protocolli di sistema standard per i prodotti OMRON.
SYSMAC WS02-PSTC1-E CX-Protocol Operation Manual	W344	Fornisce informazioni sull'utilizzo di CX-Protocol per la creazione di protocol macro, impiegate come sequenze di comunicazione nella comunicazione con dispositivi esterni.

⚠ **AVVERTENZA** La mancata lettura o comprensione delle informazioni fornite in questo manuale può fare insorgere condizioni di rischio e conseguenti lesioni personali, anche mortali, danneggiamento o funzionamento non corretto del prodotto. Leggere ogni capitolo per intero e accertarsi di avere compreso le informazioni in essa contenute e quelle correlate prima di eseguire una delle procedure o operazioni descritte.

PRECAUZIONI

In questo capitolo sono riportate le precauzioni generali per l'uso dei controllori programmabili (PLC) della serie CS/CJ e dei dispositivi collegati.

Le informazioni contenute in questo capitolo sono importanti per garantire un utilizzo sicuro e affidabile dei controllori programmabili. È necessario leggere il capitolo e comprenderne il contenuto prima di configurare o utilizzare un PLC.

1 Destinatari del manuale

Il presente manuale si rivolge al personale riportato di seguito, a cui sono richieste conoscenze in materia di sistemi elettrici (perito elettrotecnico o titolo equivalente).

- Responsabili dell'installazione di sistemi di automazione industriale.
- Responsabili della progettazione di sistemi di automazione industriale.
- Responsabili della gestione di sistemi di automazione industriale e delle relative infrastrutture.

2 Precauzioni generali

L'utente deve utilizzare il prodotto in base alle specifiche riportate nei manuali dell'operatore.

Prima di utilizzare il prodotto in condizioni non previste dal manuale o di applicarlo a sistemi di controllo nucleare, sistemi ferroviari, sistemi per aviazione, veicoli, sistemi di combustione, apparecchiature medicali, macchine da Luna Park, apparecchiature di sicurezza e qualunque altro sistema, macchina o apparecchiatura il cui utilizzo improprio possa comportare il rischio di gravi lesioni a persone e danni alla proprietà, rivolgersi al proprio rappresentante OMRON.

Accertarsi che i valori nominali e le specifiche del prodotto siano sufficienti per i sistemi, le macchine e le apparecchiature che verranno utilizzati e dotare sempre tali sistemi, macchine e apparecchiature di doppi meccanismi di sicurezza.

Il presente manuale fornisce informazioni sulla programmazione e sul funzionamento del Modulo. Si raccomanda di leggere il manuale prima di utilizzare il Modulo per la prima volta e tenerlo sempre a portata di mano come riferimento durante le operazioni.

⚠ **AVVERTENZA** È di fondamentale importanza che il PLC e tutti i relativi Moduli vengano utilizzati per lo scopo specificato e nelle condizioni specificate, in particolare in applicazioni che implicano rischi diretti o indiretti per l'incolumità delle persone. Prima di utilizzare il PLC per tali applicazioni, rivolgersi al proprio rappresentante OMRON.

3 Precauzioni per la sicurezza

⚠ **AVVERTENZA** La CPU aggiorna gli I/O anche quando il programma non è in esecuzione, ovvero anche in modalità PROGRAM. Verificare preliminarmente lo stato della sicurezza prima di modificare lo stato di qualunque parte della memoria assegnata ai Moduli di I/O, ai Moduli di I/O speciale o ai Moduli CPU bus. Qualunque modifica ai dati della memoria assegnata a un Modulo può provocare l'inattesa attivazione dei carichi collegati a tale Modulo. Una qualunque delle seguenti operazioni può determinare la modifica dello stato della memoria.

- Trasferimento di dati della memoria I/O da un dispositivo di programmazione alla CPU.
- Modifica degli attuali valori in memoria da un dispositivo di programmazione.
- Impostazione/ripristino forzato di bit da un dispositivo di programmazione.
- Trasferimento di file della memoria I/O da una schedina di memoria o dall'area di memoria file nell'area EM alla CPU.
- Trasferimento di dati della memoria I/O da un computer host o da un altro PLC collegato in rete.

⚠ **AVVERTENZA** Non tentare di aprire un Modulo in presenza di alimentazione, in quanto ciò implica il rischio di scosse elettriche.

⚠ **AVVERTENZA** Non toccare i terminali o le morsettiere quando il sistema è alimentato, in quanto ciò implica il rischio di scosse elettriche.

⚠ **AVVERTENZA** Non tentare di smontare, riparare o modificare alcun Modulo. Qualsiasi intervento in tal senso potrebbe provocare un funzionamento incorretto, incendi o scosse elettriche.

⚠ **AVVERTENZA** Applicare adeguate misure di sicurezza ai circuiti esterni (cioè, esterni al controllore programmabile), incluse quelle riportate di seguito, per garantire la massima sicurezza del sistema in caso di anomalie dovute al funzionamento incorretto del PLC o ad altri fattori esterni che influiscono sul funzionamento del PLC. Disattendere queste precauzioni potrebbe essere causa di gravi incidenti.

- I circuiti di controllo esterni devono essere dotati di circuiti di arresto di emergenza, circuiti di interblocco, circuiti di finecorsa e altre misure di sicurezza analoghe.
- Il PLC disattiva tutte le uscite quando la funzione di autodiagnostica rileva un errore o viene eseguita un'istruzione FALS (allarme di guasto grave). Come contromisura in caso di tali errori, il sistema deve essere dotato di misure di sicurezza esterne.
- Le uscite del PLC potrebbero restare attivate o disattivate in caso di deposizione elettrolitica, bruciatura dei relè di uscita o distruzione dei transistor di uscita. Come contromisura per questo problema, il sistema deve essere dotato di misure di sicurezza esterne.
- In presenza di sovraccarico o cortocircuito sull'uscita a 24 Vc.c. (alimentazione di servizio del PLC), si potrebbe verificare un abbassamento di tensione e una conseguente disattivazione delle uscite. Come contromisura per questo problema, il sistema deve essere dotato di misure di sicurezza esterne.

⚠ **Attenzione** Verificare lo stato di sicurezza prima di trasferire file di dati memorizzati nella memoria file (schedina di memoria o area di memoria file nell'area EM) all'area degli I/O (CIO) della CPU utilizzando un dispositivo periferico. In caso contrario, è possibile che i dispositivi collegati al modulo di uscita non funzionino in modo corretto, indipendentemente dalla modalità operativa della CPU.

⚠ **Attenzione** Il cliente è tenuto a implementare meccanismi di sicurezza per guasti ed errori allo scopo di garantire la sicurezza in caso di segnali errati, mancanti o anomali provocati da guasti a carico delle linee di segnale, cadute di tensione temporanee o altre cause. Un funzionamento anomalo potrebbe implicare il rischio di gravi incidenti.

⚠ **Attenzione** Le CPU CS1-H, CJ1-H, CJ1M e CS1D eseguono automaticamente il backup del programma utente e dei dati di parametro nella memoria flash quando tali dati vengono scritti nella CPU. I dati della memoria I/O (comprese le aree DM, EM e HR) non vengono tuttavia scritti nella memoria flash. Un'apposita batteria consente di conservare i dati delle aree DM, EM e HR nel caso in cui si verifichino cadute di tensione. Se si verificano errori relativi alla batteria, i dati contenuti in queste aree potrebbero risultare non corretti dopo la caduta di tensione. Se i dati delle aree DM, EM e HR vengono utilizzati per il controllo di uscite esterne, impedire l'invio di dati non corretti quando il flag di errore della batteria (A40204) è attivato.

⚠ **Attenzione** Eseguire modifiche in linea solo dopo aver verificato che l'estensione del tempo di ciclo non provoca effetti negativi. In caso contrario, i segnali di ingresso potrebbe risultare illeggibili.

⚠ **Attenzione** Verificare lo stato di sicurezza sul nodo di destinazione prima di trasferire un programma o modificare il contenuto dell'area di memoria I/O. La mancata osservanza di questa precauzione prima di procedere a tali operazioni implica il rischio di lesioni.

⚠ **Attenzione** Serrare le viti sulla morsettiera del Modulo di alimentazione c.a. applicando la coppia specificata nel manuale dell'operatore. La presenza di viti allentate può provocare bruciature o il funzionamento incorretto.

⚠ **Attenzione** Non toccare il Modulo di alimentazione mentre eroga corrente o immediatamente dopo lo spegnimento, poiché la temperatura del Modulo di alimentazione potrebbe essere tale da provocare ustioni.

⚠ **Attenzione** Prestare particolare attenzione durante il collegamento di PC o altri dispositivi periferici a un PLC su cui sia installato un modulo non isolato (CS1W-CLK12/52(-V1) o CS1W-ETN01) collegato a un alimentatore esterno. Se l'estremità a 24 V dell'alimentatore esterno viene collegata a terra e l'estremità a 0 V della periferica viene collegata a terra, si creerà un corto circuito. Durante il collegamento di una periferica a questo tipo di PLC, eseguire la messa a terra dell'estremintà a 0 V dell'alimentatore esterno oppure non eseguire affatto la messa a terra dell'alimentatore esterno.

4 Precauzioni relative all'ambiente operativo

⚠ **Attenzione** Non utilizzare il sistema di controllo nei seguenti luoghi:

- Luoghi esposti alla luce solare diretta.
- Luoghi con temperature o tassi di umidità al di fuori della gamma di valori riportata nelle specifiche.
- Luoghi soggetti a formazione di condensa a causa di considerevoli escursioni termiche.
- Luoghi esposti a gas corrosivi o infiammabili.
- Luoghi esposti a polvere (in particolare polvere metallica) o agenti salini.
- Luoghi esposti ad acqua, oli o agenti chimici.
- Luoghi soggetti a urti o vibrazioni.

⚠ **Attenzione** Applicare soluzioni di sicurezza adeguate e sufficienti quando si installano sistemi nei seguenti luoghi:

- Luoghi soggetti a elettricità statica o altre forme di disturbi.
- Luoghi in cui sono presenti forti campi elettromagnetici.
- Luoghi potenzialmente esposti a radioattività.
- Luoghi prossimi a linee elettriche.

⚠ **Attenzione** L'ambiente in cui opera il PLC può avere un grande impatto sulla durata e sull'affidabilità del sistema. L'utilizzo in ambienti operativi non appropriati può essere causa di funzionamento incorretto, guasti e altri problemi non prevedibili. Accertarsi che l'ambiente operativo rispetti le condizioni richieste per l'installazione e che tali condizioni siano mantenute per l'intera durata di esercizio del sistema.

5 Precauzioni relative all'applicazione

Osservare le seguenti precauzioni nell'uso del PLC.

- Per la programmazione di più task, utilizzare CX-Programmer, il software di programmazione per Windows. La Console di programmazione può essere utilizzata per programmare un solo task ciclico e task ad interrupt. È tuttavia possibile utilizzare una Console di programmazione per modificare programmi composti da più task creati con CX-Programmer.

⚠ **AVVERTENZA** Attenersi sempre alle seguenti precauzioni. Il mancato rispetto di tali precauzioni può essere causa di lesioni gravi, anche mortali.

- Durante l'installazione dei Moduli, effettuare sempre un collegamento a terra con una resistenza di 100 Ω o inferiore. Il mancato collegamento a terra a una resistenza di 100 Ω o inferiore potrebbe determinare scosse elettriche.
- Se si collegano in cortocircuito i terminale di messa a terra della linea (LG) e di messa a terra (GR) sul Modulo di alimentazione, eseguire un collegamento a terra con una resistenza di 100 Ω o inferiore.
- Spegnere sempre il PLC e scollegare l'alimentazione prima di eseguire una delle operazioni riportate di seguito. La mancata interruzione dell'alimentazione, comporta il rischio di scosse elettriche o il funzionamento incorretto.
  - Montaggio o smontaggio di Moduli di alimentazione, Moduli di I/O, CPU, schede interne o altri Moduli.
  - Assemblaggio di Moduli.
  - Impostazione di DIP switch o di selettori rotanti.
  - Collegamento di cavi o cablaggio del sistema.
  - Collegamento e scollegamento di connettori.

⚠ **Attenzione** Il mancato rispetto delle seguenti precauzioni può causare il funzionamento incorretto del PLC o del sistema o danni al PLC e ai relativi Moduli. Seguire sempre tali precauzioni.

- Una copia di backup del programma utente e dei dati dell'area dei parametri nelle CPU CS1-H, CS1D, CJ1-H e CJ1M viene salvata nella memoria flash integrata. Durante l'esecuzione del backup, l'indicatore BKUP posto sulla parte anteriore della CPU si accende. Non spegnere la CPU quando l'indicatore BKUP è acceso. Se si interrompe l'alimentazione, il backup dei dati non verrà eseguito.
- Quando si utilizza una CPU CS1 per la prima volta, prima di procedere alla programmazione installare la batteria CS1W-BAT1 fornita con la CPU e cancellare tutte le aree di memoria utilizzando un dispositivo di programmazione. Quando si utilizza l'orologio interno, accendere il sistema dopo aver installato la batteria e impostare l'orologio tramite un dispositivo di programmazione o utilizzando l'istruzione DATE(735). L'orologio non verrà avviato finché non viene impostata l'ora.
- Al momento dell'acquisto, il PLC è configurato in modo tale che la CPU venga avviata nella modalità operativa impostata tramite il selettore di modalità della Console di programmazione. Se non si collega una Console di programmazione, le CPU CS1 della serie CS verranno avviate in modalità PROGRAM, mentre le CPU CS1-H, CS1D, CJ1, CJ1-H o CJ1M verranno avviate in modalità RUN, entrando immediatamente in funzione. Accertarsi che i Moduli non entrino in funzione senza avere prima verificato le condizioni di sicurezza.

- Quando si crea un file AUTOEXEC.IOM mediante un dispositivo di programmazione, quale una Console di programmazione o il software CX-Programmer, per il trasferimento automatico dei dati all'avvio, impostare il primo indirizzo di scrittura su D20000 e verificare che le dimensioni dei dati non superino la capacità dell'area DM. Quando all'avvio viene letto il file di dati dalla schedina di memoria, i dati vengono scritti nella CPU a partire dall'indirizzo D20000, anche se al momento della creazione del file AUTOEXEC.IOM è stato specificato un indirizzo diverso. Inoltre, se si supera la capacità dell'area DM, e tale possibilità può verificarsi quando si utilizza CX-Programmer, i dati rimanenti verranno scritti nell'area EM.
- Accendere sempre il PLC prima di accendere il sistema di controllo. Se il PLC viene acceso dopo il sistema di controllo, potrebbero verificarsi errori di segnale temporanei a livello del sistema di controllo, in quanto i terminali di uscita sui Moduli di uscita c.c. e altri Moduli vengono momentaneamente sollecitati quando si accende il PLC.
- Il cliente è tenuto a implementare meccanismi di sicurezza per guasti ed errori allo scopo di garantire la sicurezza nel caso in cui le uscite dei Moduli di uscita rimangano attivate a seguito di guasti dei circuiti interni a carico di relè, transistor e altri elementi.
- Il cliente è tenuto a implementare meccanismi di sicurezza per guasti ed errori allo scopo di garantire la sicurezza in caso di segnali errati, mancanti o anomali provocati da guasti a carico delle linee di segnale, cadute di tensione temporanee o altre cause.
- È responsabilità del cliente dotare i circuiti esterni, ovvero esterni al controllore programmabile, di circuiti di interblocco, circuiti di finecorsa e altre misure di sicurezza analoghe.
- Non spegnere il PLC durante il trasferimento di dati. In particolare, non spegnere il PLC durante le operazioni di lettura o scrittura su una schedina di memoria e non rimuovere la schedina mentre l'indicatore BUSY è acceso. Per rimuovere una schedina di memoria, premere innanzitutto l'interruttore di alimentazione della schedina, quindi attendere che l'indicatore BUSY si spenga prima di rimuoverla.
- Se il bit di ritenzione IOM è impostato su ON e si passa dalla modalità operativa RUN o MONITOR alla modalità PROGRAM, le uscite del PLC non vengono disattivate e mantengono lo stato precedente. Accertarsi che i carichi esterni non instaurino condizioni pericolose quando ciò accade. Se il funzionamento viene interrotto a causa di un errore fatale, inclusi gli errori generati dall'istruzione FALS(007), tutte le uscite del Modulo di uscita vengono disattivate e viene mantenuto solo lo stato delle uscite interne.
- L'integrità del contenuto delle aree DM, EM e HR della CPU è assicurata da una batteria di backup. Se la batteria si scarica, i dati potrebbero andare persi. Adottare le soluzioni adeguate a livello di programma utilizzando il flag di errore della batteria (A40204) per la reinizializzazione dei dati o la segnalazione delle azioni da intraprendere nel caso in cui la carica della batteria si esaurisca.
- Quando si fornisce corrente a 200-240 Vc.a. con un PLC della serie CS, rimuovere sempre il ponticello di metallo dai terminali del selettore di tensione del Modulo di alimentazione (tale operazione non è necessaria per Moduli di alimentazione con gamme di tensione più ampie). Se il ponticello di metallo rimane collegato durante l'alimentazione a 200-240 Vc.a., il prodotto verrà distrutto.
- Utilizzare sempre tensioni di alimentazione specificate nei manuali dell'operatore. Una tensione errata può provocare un funzionamento incorretto o bruciature.

- Adottare le misure necessarie per garantire che il sistema sia sempre alimentato nel rispetto delle specifiche di tensione e frequenza nominali. Prestare particolare attenzione in ambienti in cui l'alimentazione è instabile. Un'alimentazione non adeguata può comportare un funzionamento incorretto.
- Installare interruttori esterni o altri dispositivi di sicurezza per evitare cortocircuiti nelle aree di cablaggio esterno. Misure insufficienti di protezione da cortocircuiti potrebbero causare bruciature.
- Non applicare ai Moduli di ingresso tensioni superiori alla tensione di ingresso nominale. Tensioni eccessivamente alte potrebbero essere causa di bruciature.
- Non applicare tensioni o collegare carichi ai Moduli di uscita superiori alla corrente di carico massima. Tensioni o carichi eccessivamente elevati potrebbero essere causa di bruciature.
- Separare il terminale di messa a terra della linea (LG) dal terminale di messa a terra funzionale (GR) sul Modulo di alimentazione prima di eseguire test della tensione di tenuta o test della resistenza di isolamento. La mancata separazione dei terminali potrebbe causare danni.
- Installare i Moduli in modo appropriato, seguendo le istruzioni riportate nei manuali dell'operatore. L'installazione errata dei Moduli può comportare un funzionamento incorretto.
- Con i PLC della serie CS, accertarsi che tutte le viti di montaggio del Modulo e del rack siano serrate rispettando la coppia specificata nei relativi manuali. Una coppia di serraggio non appropriata può comportare un funzionamento incorretto.
- Accertarsi che tutte le viti dei terminali e le viti dei connettori dei cavi siano serrate rispettando la coppia specificata nei relativi manuali. Una coppia di serraggio non appropriata può comportare un funzionamento incorretto.
- Durante il cablaggio, lasciare l'etichetta attaccata al Modulo. La rimozione dell'etichetta può comportare la penetrazione di materiale estraneo nel Modulo e il conseguente funzionamento incorretto.
- Una volta completato il cablaggio, rimuovere l'etichetta per garantire un'appropriata dissipazione del calore. Se non si rimuove l'etichetta, il Modulo potrebbe non funzionare correttamente.
- Quando si procede al cablaggio, utilizzare terminali a crimpare. Non collegare direttamente ai terminali fili scoperti. Il collegamento diretto di fili scoperti può causare bruciature.
- Accertarsi di cablare correttamente tutti i collegamenti.
- Verificare a fondo l'intero cablaggio e le impostazioni degli interruttori prima di alimentare il Modulo. Un cablaggio errato può essere causa di bruciature.
- Installare i Moduli solo dopo aver verificato tutte le morsettiere e i connettori.
- Accertarsi che le morsettiere, i Moduli di memoria, le prolunghe e altri componenti dotati di dispositivi di bloccaggio siano correttamente bloccati in posizione. L'errato bloccaggio di questi componenti può causare un funzionamento incorretto.
- Verificare le impostazioni degli interruttori, il contenuto dell'area DM e ogni altro prerequisito prima di mettere in funzione il sistema. L'avvio in presenza di impostazioni o dati non corretti può provocare un funzionamento imprevisto.

- Verificare la corretta esecuzione del programma utente prima di eseguirlo sul Modulo. La mancata verifica del programma può provocare un funzionamento imprevisto.
- Prima di eseguire le operazioni riportate di seguito, accertarsi che non abbiano effetti negativi sul sistema. Disattendere questa precauzione potrebbe dare luogo a un funzionamento imprevisto.
  - Modifica della modalità operativa del PLC.
  - Impostazione/ripristino forzato di qualunque bit in memoria.
  - Modifica del valore attuale di qualsiasi canale o valore impostato in memoria.
- Non tirare o piegare i cavi oltre il limite di resistenza naturale. Ciò potrebbe provocarne la rottura.
- Non appoggiare alcun oggetto sui cavi. Ciò potrebbe provocarne la rottura.
- Non utilizzare cavi RS-232C per personal computer di terze parti. Utilizzare sempre i cavi speciali elencati in questo manuale o assemblare i cavi in conformità alle specifiche indicate. L'impiego di cavi di terze parti potrebbe causare danni ai dispositivi esterni o alla CPU.
- Non collegare mai il pin 6 (alimentazione a 5 V) della porta RS-232C di una CPU a un dispositivo diverso da un adattatore NT-AL001 o CJ1W-CIF11. Il dispositivo esterno o la CPU potrebbe subire danni.
- Quando si sostituiscono componenti, accertarsi sempre che le specifiche tecniche del nuovo componente siano appropriate. Disattendere questa precauzione può causare un funzionamento incorretto o bruciature.
- Prima di toccare un Modulo, toccare un oggetto metallico con messa a terra per scaricare l'elettricità statica accumulata. Disattendere questa precauzione può causare un funzionamento incorretto o danneggiare il Modulo.
- Quando si trasportano o immagazzinano schede di circuiti stampati, coprirle sempre con materiale antistatico per proteggerle dall'elettricità statica e mantenere la temperatura appropriata.
- Non toccare le schede di circuiti stampati o i componenti montati sulle schede a mani nude. Sulle schede vi sono contatti appuntiti ed altre parti che, se toccate incautamente, potrebbero provocare lesioni.
- Non cortocircuitare i terminali della batteria né caricare, smontare, scaldare eccessivamente o incenerire la batteria. Non sottoporre la batteria a forti urti. Ciò potrebbe provocare dispersione, rottura, emissione di calore o combustione della batteria. Eliminare le batterie che hanno subito forti urti, ad esempio cadendo sul pavimento, in quanto potrebbero verificarsi perdite di elettroliti durante l'uso.
- Gli standard UL richiedono che le batterie vengano sostituite solo da tecnici esperti. Non consentire la sostituzione delle batterie da parte di personale non qualificato.
- Nei PLC della serie CJ, i dispositivi scorrevoli posti sulla parte superiore e inferiore del Modulo di alimentazione, della CPU, dei Moduli di I/O, dei Moduli di I/O speciale e dei Moduli CPU bus devono essere bloccati, facendoli scorrere finché non scattano in posizione. Se i dispositivi scorrevoli non vengono bloccati, il Modulo non funzionerà in modo corretto.
- Nei PLC della serie CJ, collegare sempre la piastrina di blocco al Modulo sul lato destro del PLC. Senza piastrina di blocco il PLC non funzionerà in modo corretto.

- L'impostazione di tabelle data link o parametri non corretti può dare luogo a un funzionamento imprevisto. Anche se le tabelle data link e i parametri sono stati impostati correttamente, verificare che il sistema controllato non ne sia influenzato negativamente prima di avviare o interrompere i data link.
- Quando le tabelle di routing vengono trasferite da un dispositivo di programmazione alla CPU, i Moduli CPU bus vengono riavviati. Il riavvio di tali Moduli è necessario per la lettura e l'abilitazione delle nuove tabelle di routing. Verificare che il sistema non evidenzi problemi prima di consentire la reimpostazione dei Moduli CPU bus.

6 Conformità alle direttive dell'Unione Europea

6-1 Direttive applicabili

- Direttive sulla compatibilità elettromagnetica (EMC)
- Direttiva per le basse tensioni (LVD)

6-2 Principi

Direttive sulla compatibilità elettromagnetica (EMC)

I dispositivi OMRON conformi alle Direttive dell'Unione Europea sono altresì conformi agli standard EMC, in modo da poter essere facilmente integrati con altri dispositivi o macchine complesse. La conformità dei singoli prodotti agli standard EMC è stata verificata (vedere nota). Tuttavia, la conformità del prodotto agli standard, una volta installato nel sistema del cliente, deve essere verificata dal cliente stesso.

Le prestazioni relative agli standard EMC dei dispositivi OMRON conformi alle Direttive dell'Unione Europea variano a seconda della configurazione, del cablaggio e di altre condizioni dell'apparecchiatura o del pannello di controllo su cui i dispositivi OMRON sono installati. Pertanto, sarà cura del cliente effettuare i controlli finali per accertare che tali dispositivi e la macchina nel suo complesso siano conformi agli standard EMC.

Nota Gli standard di compatibilità elettromagnetica applicabili sono i seguenti:

EMS (suscettibilità elettromagnetica):
Serie CS: EN 61131-2 e EN 61000-6-2
Serie CJ: EN61000-6-2
EMI (interferenza elettromagnetica):
EN61000-6-4
(Emissione irradiata: norme 10 m)

Direttiva per le basse tensioni (LVD)

Accertarsi sempre che i dispositivi che operano nella gamma di tensioni 50-1.000 Vc.a. o 75-1.500 Vc.c. soddisfino gli standard di sicurezza per il PLC (EN 61131-2).

6-3 Conformità alle direttive dell'Unione Europea

I PLC della serie CS/CJ sono conformi alle Direttive dell'Unione Europea. Per garantire che la macchina o il dispositivo in cui viene utilizzato il PLC della serie CS/CJ sia conforme alle Direttive dell'Unione Europea, è necessario soddisfare i seguenti requisiti di installazione del PLC:

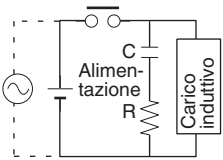
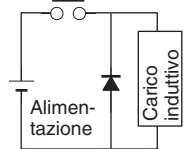
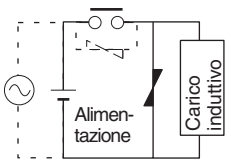
1,2,3... 1. Il PLC della serie CS/CJ deve essere installato in un pannello di controllo.

2. È necessario utilizzare un isolamento rinforzato o un doppio isolamento per gli alimentatori c.c. collegati ai Moduli di alimentazione c.c. e ai Moduli di I/O.
3. I PLC della serie CS/CJ conformi alle Direttive dell'Unione Europea sono altresì conformi agli standard generici sulle emissioni (EN61000-6-4). Le caratteristiche per le emissioni irradiate possono variare a in base alla configurazione del pannello di controllo utilizzato, agli altri dispositivi collegati al pannello di controllo, al cablaggio e ad altre condizioni specifiche. È pertanto necessario verificare che la macchina nel suo complesso o l'apparecchiatura utilizzata sia conforme alle Direttive dell'Unione Europea.

6-4 Metodi di riduzione dei disturbi nelle uscite a relè

I PLC della serie CS/CJ sono conformi agli standard generici sulle emissioni (EN61000-6-4) delle Direttive EMC. Tuttavia, i disturbi generati dalla commutazione delle uscite a relè potrebbero non risultare conformi a questi standard. Qualora ciò si verifichi, è necessario applicare un filtro antidisturbo dal lato carico o adottare altre soluzioni equivalenti esterne al PLC.

Le soluzioni da adottare per soddisfare i requisiti posti dagli standard possono variare a seconda dei dispositivi collegati sul lato carico, del cablaggio, della configurazione delle macchine e così via. Di seguito sono riportati alcuni esempi di soluzioni per la riduzione dei disturbi generati.

Contromisure

Per ulteriori dettagli, fare riferimento alla norma EN61000-6-4.

Non è necessaria alcuna contromisura per la riduzione dei disturbi se la frequenza di commutazione del carico per l'intero sistema, incluso il PLC, è inferiore a 5 volte al minuto.

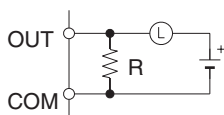
È necessario adottare contromisure per la riduzione dei disturbi se la frequenza di commutazione del carico per l'intero sistema, incluso il PLC, è superiore a 5 volte al minuto.

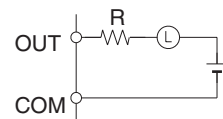
Esempi di contromisure

Quando si commuta un carico induttivo, collegare un dispositivo di protezione da sovracorrente, diodi e così via in parallelo con il carico o contatto come mostrato di seguito.

Circuito	Corrente		Caratteristica	Elemento necessario
	c.a.	c.c.		
Metodo CR	Sì	Sì	Se il carico è un relè o solenoide, si determina un ritardo tra l'apertura del circuito e il ripristino del carico. Se la tensione di alimentazione è 24 o 48 V, collegare il dispositivo di protezione da sovracorrente in parallelo con il carico. Se la tensione di alimentazione è da 100 a 200 V, inserire il dispositivo di protezione da sovracorrente tra i contatti.	La capacità del condensatore deve essere di 1-0,5 µF per una corrente di contatto pari a 1 A e la resistenza del resistore deve essere di 0,5-1 Ω per una tensione di contatto pari a 1 V. Tuttavia, questi valori possono variare in base al carico e alle caratteristiche del relè. Definire tali valori sulla base di test, tenendo in considerazione il fatto che il condensatore sopprime la scarica della scintilla quando i contatti vengono separati, mentre il resistore limita il flusso di corrente verso il carico quando il circuito viene richiuso. La rigidità dielettrica del condensatore deve essere di 200-300 V. Se il circuito è un circuito c.a., utilizzare un condensatore senza polarità.
Metodo con diodo	No	Sì	L'energia accumulata nella bobina viene trasformata in corrente dal diodo collegato in parallelo al carico, quindi la corrente che transita nella bobina viene assorbita e convertita in calore dalla resistenza del carico induttivo. Il ritardo tra l'apertura del circuito e il ripristino del carico indotto da questo metodo è più lungo di quello ottenuto con il metodo CR (condensatore-resistore).	La rigidità dielettrica inversa del diodo deve essere almeno 10 volte superiore alla tensione del circuito. La corrente diretta del diodo deve essere pari o superiore alla corrente di carico. La rigidità dielettrica inversa del diodo deve essere da due a tre volte superiore alla tensione di alimentazione se ai circuiti elettronici con basse tensioni è applicato un dispositivo di protezione da sovracorrente.
Metodo con varistore	Sì	Sì	Il metodo con variatore impedisce l'imposizione di tensioni elevate tra i contatti sfruttando la tensione costante caratteristica del variatore. Si determina un ritardo tra l'apertura del circuito e il ripristino del carico. Se la tensione di alimentazione è 24 o 48 V, collegare il variatore in parallelo con il carico. Se la tensione di alimentazione è da100 a 200 V, inserire il varistore tra i contatti.	---

Quando si commuta un carico con una forte corrente di picco, come in una lampada a incandescenza, ridurre la corrente come illustrato di seguito.

Soluzione 1

Generando una corrente di riposo pari a circa un terzo del valore nominale della lampada a incandescenza

Soluzione 2

Utilizzando un resistore per limitare il flusso di corrente

CAPITOLO 1
Funzionamento della CPU

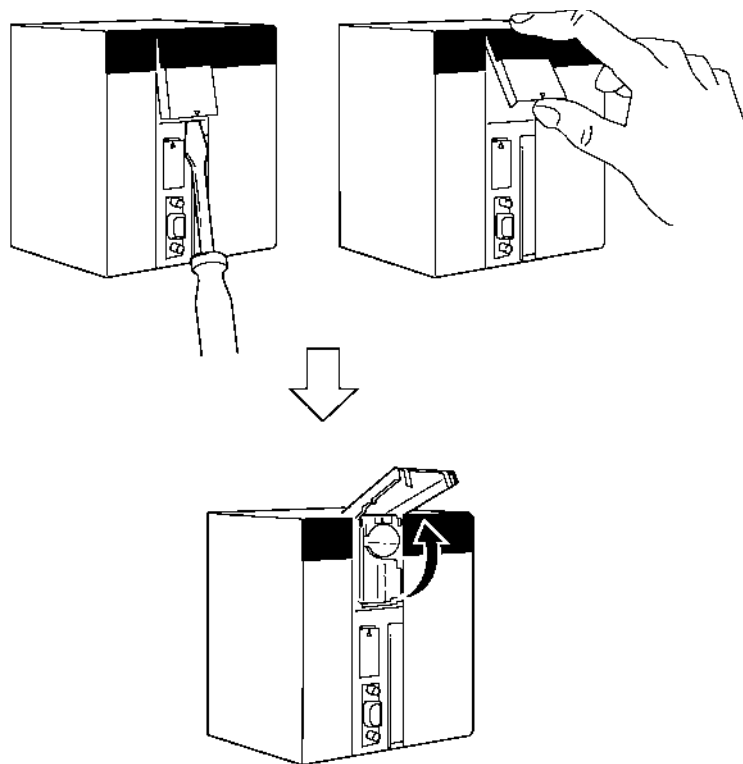
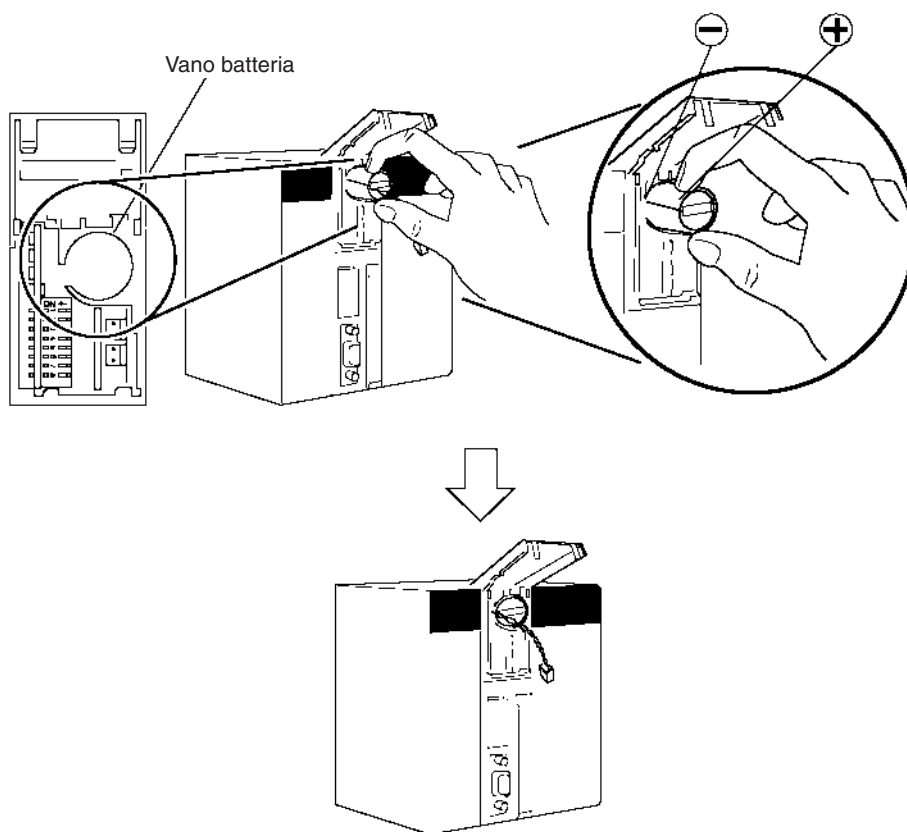
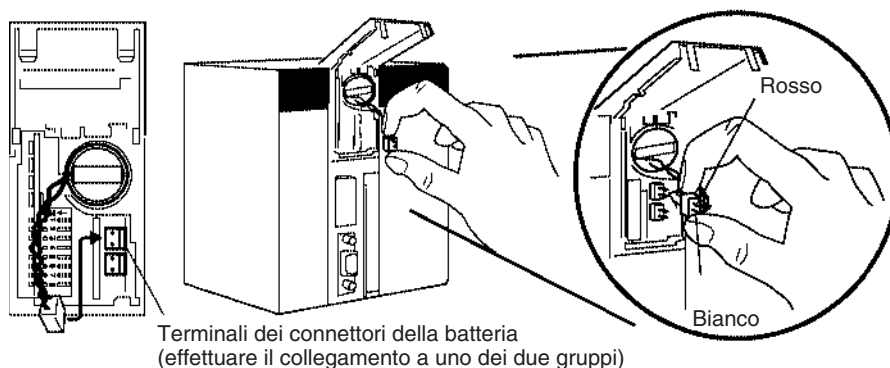
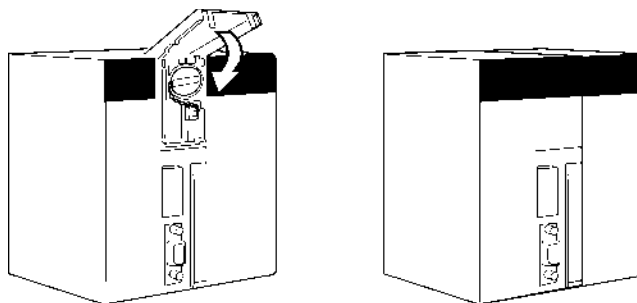
In questo capitolo vengono descritti la struttura di base e il funzionamento della CPU.

1-1 Configurazione iniziale (solo CPU CS1)

Installazione della batteria

Prima di utilizzare una CPU CS1, è necessario installarvi la batteria attenendosi alla seguente procedura.

1,2,3...

1. Per accedere all'alloggiamento della batteria, inserire la punta di un cacciavite a taglio nella fessura alla base dell'alloggiamento e fare leva verso l'alto per aprire il coperchio.

2. Tenere la batteria con il cavo rivolto verso l'esterno e inserirla nell'alloggiamento.

3. Collegare il connettore della batteria agli appositi terminali. Collegare il filo rosso al terminale superiore e il filo bianco al terminale inferiore. È possibile utilizzare uno qualsiasi dei due gruppi di terminali per i connettori della batteria disponibili. La batteria funzionerà in modo corretto indipendentemente dal gruppo di terminali utilizzato.

4. Ripiegare il cavo e chiudere il coperchio.

Cancellazione della memoria

Dopo aver installato la batteria, cancellare la memoria utilizzando la procedura appropriata per inizializzare la RAM all'interno della CPU.

Console di programmazione

Attenersi alla procedura descritta utilizzando una Console di programmazione.

Nota Quando si cancella la memoria mediante una Console di programmazione, non è possibile specificare più di un task ciclico. È possibile specificare un task ciclico e un task ad interrupt oppure un task ciclico senza task ad interrupt. Per ulteriori informazioni sull'operazione di cancellazione della memoria, consultare il *Manuale dell'operatore*. Per ulteriori informazioni sui task, fare riferimento a *CAPITOLO 1 Funzionamento della CPU* e *CAPITOLO 4 Task*.

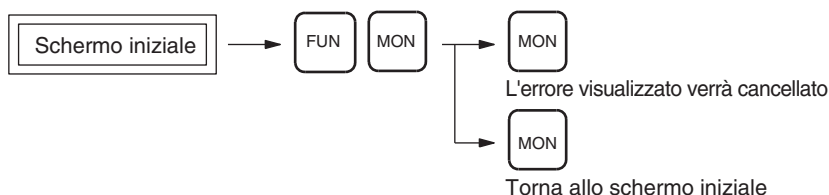
CX-Programmer

È possibile cancellare la memoria anche con CX-Programmer. Per informazioni sulla procedura, consultare il *Manuale dell'operatore* di CX-Programmer.

Cancellazione degli errori

Dopo aver cancellato la memoria, cancellare gli eventuali errori della CPU, compreso l'errore di batteria scarica.

Console di programmazione

Attenersi alla procedura descritta utilizzando una Console di programmazione.

CX-Programmer

È possibile cancellare gli errori anche con CX-Programmer. Per informazioni sulla procedura, consultare il *Manuale dell'operatore* di CX-Programmer.

Nota Quando è installata una scheda interna, è possibile che l'eventuale errore della scheda interna relativo alla tabella di routing persista anche dopo la cancellazione dell'errore con CX-Programmer (lo stato di A42407 resterà ON per una scheda di comunicazione seriale). In tal caso, spegnere e riaccendere il sistema o riavviare la scheda interna, quindi cancellare nuovamente l'errore.

1-2 Utilizzo dell'orologio interno (solo CPU CS1)

Quando in una CPU della serie CS viene installata la batteria, l'orologio interno della CPU viene impostato su "anno 00, mese 01, giorno 01 (00-01-01), 00 ore, 00 minuti, 00 secondi (00:00:00) e domenica (SUN)".

Quando si utilizza l'orologio interno, ripristinare l'alimentazione dopo aver installato la batteria e 1) utilizzare un dispositivo di programmazione (Console di programmazione o CX-Programmer) per impostare l'ora, 2) eseguire l'istruzione CLOCK ADJUSTMENT (DATE) oppure 3) inviare un comando FINS per avviare l'orologio interno a partire dalla data e ora correnti.

Di seguito è illustrata la procedura utilizzata per impostare l'orologio interno con una Console di programmazione.

Sequenza di tasti

1-3 Struttura interna della CPU

1-3-1 Informazioni generali

Lo schema seguente illustra la struttura interna della CPU.

Programma utente

Il programma utente è composto da un massimo di 288 task di programma, compresi i task ad interrupt. I task vengono trasferiti alla CPU dal software di programmazione CX-Programmer.

Esistono due tipi di task. Il primo è un task ciclico che viene eseguito una volta per ciclo (fino a un massimo di 32) e l'altro è un task ad interrupt che viene eseguito solo quando si verificano le condizioni di interrupt (fino a un massimo di 256). I task ciclici vengono eseguiti in ordine numerico.

Nota

1. Con una CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt in maniera ciclica, in modo analogo ai task ciclici. Tali task vengono definiti "task ciclici supplementari". Il numero totale di task che è possibile eseguire ciclicamente deve essere inferiore o pari a 288.
2. Quando si utilizza CX-Programmer, utilizzare la versione 2.1 o successiva per una CPU CS1-H o CJ1-H e la versione 3.0 o successiva per una CPU CJ1M (fatta eccezione per i modelli di base) o CS1D per sistemi a due CPU. Se si impiega una CPU CJ1M di base (CJ1M-CPU11/CPU21), utilizzare CX-Programmer versione 3.3 o successiva. Se si impiega una CPU CS1-H, CJ1-H, CJ1M o CS1D versione 2.0 o successiva, utilizzare CX-Programmer versione 4.0 o successiva.

Le istruzioni di programma eseguono operazioni di lettura e scrittura nella memoria I/O e vengono eseguite in sequenza a partire dall'inizio del programma. Una volta eseguiti tutti i task ciclici, gli I/O di tutti i Moduli vengono aggiornati e il ciclo si ripete nuovamente a partire dal numero di task ciclico più basso.

Per ulteriori informazioni sull'aggiornamento degli I/O, consultare il manuale *CS/CJ Series Operation Manual*.

Memoria I/O

La memoria I/O è l'area della RAM utilizzata per la lettura e scrittura da un programma utente. È costituita da un'area che viene cancellata quando l'alimentazione viene interrotta e ripristinata e da un'altra area deputata alla conservazione dei dati.

La memoria I/O è inoltre suddivisa in due aree: una utilizzata per lo scambio dei dati con tutti i Moduli e un'altra ad esclusivo uso interno. I dati vengono scambiati con tutti i Moduli una volta ogni ciclo di esecuzione del programma, nonché quando vengono eseguite istruzioni specifiche.

Configurazione del PLC

La configurazione del PLC viene utilizzata per specificare diverse impostazioni iniziali o altre impostazioni tramite parametri software.

DIP switch

I DIP switch vengono utilizzati per configurare impostazioni iniziali o altre impostazioni tramite interruttori hardware.

schedine di memoria

Le schedine di memoria vengono utilizzate all'occorrenza per memorizzare dati quali programmi, dati della memoria I/O, la configurazione del PLC e i commenti degli I/O creati mediante dispositivi di programmazione. I programmi e varie impostazioni di sistema possono essere automaticamente letti dalla schedina di memoria e scritti nella CPU all'accensione del sistema (trasferimento automatico all'avvio).

Memoria flash (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)

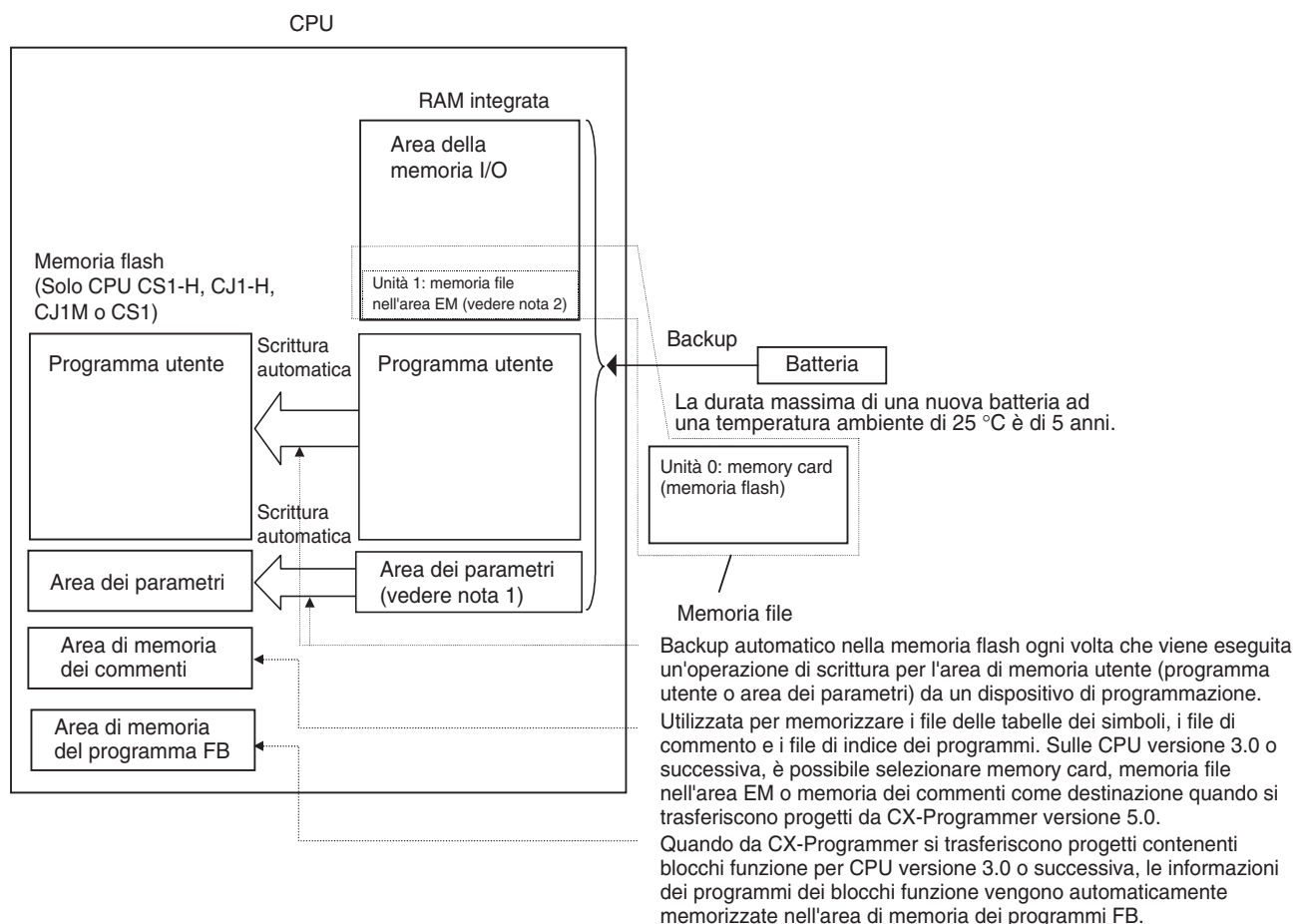
Per quanto riguarda le CPU CS1-H, CJ1-H, CJ1M o CS1D, ogni volta che l'utente procede alla scrittura di dati nella CPU, una copia di backup del programma utente e dei dati dell'area dei parametri, quale la configurazione del PLC, viene automaticamente salvata nella memoria flash integrata della CPU. In tal modo è possibile ottenere il funzionamento senza batteria e senza utilizzare una schedina di memoria. Si tenga presente che in assenza della batteria non viene eseguito il backup della memoria I/O, compresa la maggior parte dei dati dell'area DM.

1-3-2 Schema a blocchi della memoria della CPU

La memoria (RAM) delle CPU della serie CS/CJ è costituita dai seguenti blocchi:

- Area dei parametri (configurazione del PLC, tabella di I/O registrati, tabella di routing e impostazioni del Modulo CPU bus)
- Aree della memoria I/O
- Programma utente

L'integrità dei dati dell'area dei parametri e delle aree della memoria I/O è assicurata da una batteria di backup (serie CS: CS1W-BAT01, CJ1-H: CPM2A-BAT01). Se la batteria si scarica, i dati vengono persi.

Le CPU CS1-H, CJ1-H, CJ1M o CS1D sono invece dotate di una memoria flash integrata per il backup dei dati. Il backup dei dati del programma utente e dell'area dei parametri viene eseguito automaticamente nella memoria flash integrata ogni volta che l'utente procede alla scrittura di dati nella CPU mediante un dispositivo di programmazione (ad esempio, CX-Programmer o una Console di programmazione), incluse le operazioni di trasferimento di dati, modifica in linea, trasferimento da schedine di memoria e così via. In pratica, i dati dell'area dei parametri e il programma utente non vanno perduti anche in caso di scaricamento della batteria.

Nota

1. È possibile proteggere da scrittura l'area dei parametri e il programma utente, vale a dire la memoria utente, impostando su ON il pin 1 del DIP switch sul pannello frontale della CPU.
2. La memoria file nell'area EM è una porzione dell'area EM convertita in memoria file, come specificato nella configurazione del PLC. Ai fini della memorizzazione di file di dati e di programma, è possibile utilizzare tutti i banchi EM dal banco specificato alla fine dell'area EM esclusivamente come memoria file.
3. Prima di utilizzare una CPU CS1 per la prima volta, accertarsi di aver installato la batteria fornita (CS1W-BAT01). Dopo l'installazione della batteria, utilizzare un dispositivo di programmazione per cancellare la RAM del PLC (area dei parametri, area della memoria I/O e programma utente).
4. La batteria delle CPU CS1-H, CJ1, CJ1-H, CJ1M o CS1D viene già installata in fabbrica. Non è quindi necessario cancellare la memoria o impostare l'ora.
5. Durante la scrittura di dati nella memoria flash, l'indicatore BKUP posto sulla parte anteriore della CPU si illumina. Non spegnere la CPU fino al termine dell'operazione di backup, ovvero finché l'indicatore BKUP non si sarà spento. Per ulteriori informazioni, fare riferimento alla sezione *6-6-11 Memoria flash*.

1-4 Modalità operative

1-4-1 Descrizione delle modalità operative

La CPU presenta le modalità operative descritte di seguito. Tali modalità consentono di controllare l'intero programma utente e sono comuni a tutti i task.

Modalità PROGRAM

In modalità PROGRAM l'esecuzione del programma viene interrotta e l'indicatore RUN non è illuminato. Questa modalità viene utilizzata per la modifica del programma o altre operazioni preparatorie quali:

- Registrazione della tabella di I/O
- Modifica della configurazione del PLC e di altre impostazioni
- Trasferimento e verifica dei programmi
- Impostazione e ripristino forzato di bit per la verifica del cablaggio e dell'assegnazione dei bit

In questa modalità, tutti i task ciclici e ad interrupt hanno stato non eseguibile (INI), ossia vengono arrestati. Per ulteriori dettagli sui task, fare riferimento alla sezione *1-6 Descrizione dei task*.
L'aggiornamento degli I/O viene eseguito in modalità PROGRAM. Per informazioni sull'aggiornamento degli I/O, consultare il *Manuale dell'operatore*.

⚠ AVVERTENZA La CPU aggiorna gli I/O anche quando il programma non è in esecuzione, ovvero anche in modalità PROGRAM. Verificare preliminarmente lo stato della sicurezza prima di modificare lo stato di qualunque parte della memoria assegnata ai Moduli di I/O, ai Moduli di I/O speciale o ai Moduli CPU bus. Qualunque modifica ai dati della memoria assegnata a un Modulo può provocare l'inattesa attivazione dei carichi collegati a tale Modulo. Una qualunque delle seguenti operazioni può determinare la modifica dello stato della memoria.

- Trasferimento di dati della memoria I/O da un dispositivo di programmazione alla CPU.
- Modifica degli attuali valori in memoria da un dispositivo di programmazione.
- Impostazione/ripristino forzato di bit da un dispositivo di programmazione.
- Trasferimento di file della memoria I/O da una schedina di memoria o dall'area di memoria file nell'area EM alla CPU.
- Trasferimento di dati della memoria I/O da un computer host o da un altro PLC collegato in rete.

Modalità MONITOR

Utilizzando i dispositivi di programmazione è possibile effettuare le operazioni descritte di seguito mentre il programma è in esecuzione in modalità MONITOR. L'indicatore RUN risulterà acceso. Questa modalità viene utilizzata per esecuzioni di prova o altre rettifiche.

- Modifica in linea
- Impostazione e ripristino forzato di bit
- Modifica di valori nella memoria I/O

In questa modalità, i task ciclici specificati per l'esecuzione all'avvio (vedere la nota) e quelli resi eseguibili da TKON(820) vengono eseguiti quando l'esecuzione del programma raggiunge il numero di task corrispondente. I task ad interrupt vengono eseguiti quando si verificano le rispettive condizioni di interrupt.

Nota I task eseguiti all'avvio vengono specificati nelle proprietà del programma tramite CX-Programmer.

Modalità RUN

Questa modalità viene utilizzata per la normale esecuzione del programma. L'indicatore RUN risulterà acceso. Alcune operazioni da dispositivo di programmazione, quali la modifica in linea, l'impostazione e il ripristino forzato e la modifica di valori della memoria I/O, sono disabilitate in questa modalità. Sono tuttavia abilitate altre operazioni come il monitoraggio dello stato di esecuzione del programma (monitoraggio programmi e monitoraggio memoria I/O).

Utilizzare questa modalità per il normale funzionamento del sistema. L'esecuzione dei task avviene in modo analogo all'esecuzione in modalità MONITOR.

Per informazioni dettagliate sulle operazioni disponibili in ciascuna modalità operativa, vedere il capitolo *10-2 Modalità operative della CPU* nel *Manuale dell'operatore*.

1-4-2 Inizializzazione della memoria I/O

La tabella che segue mostra quali aree dati vengono cancellate quando la modalità operativa passa da PROGRAM a RUN/MONITOR o viceversa.

Cambio modalità	Aree senza ritenzione (nota 1)	Aree con ritenzione (nota 2)
RUN/MONITOR → PROGRAM	Cancellata (nota 3)	Mantenuta
PROGRAM → RUN/MONITOR	Cancellata (nota 3)	Mantenuta
RUN ↔ MONITOR	Mantenuta	Mantenuta

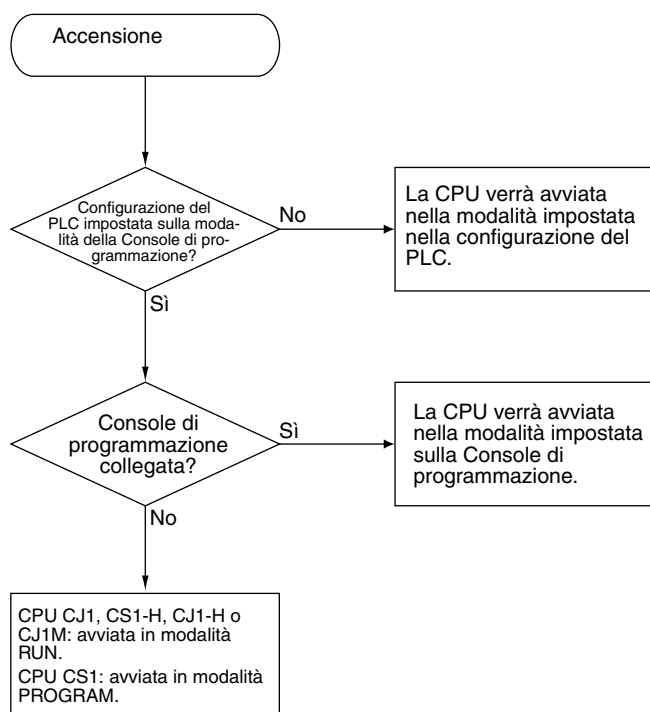
Nota

1. Aree senza ritenzione: area CIO, area di lavoro, valori attuali del temporizzatore, flag di completamento del temporizzatore, registri indice, registri dati, flag dei task e flag di condizione
   (Le informazioni di stato di alcuni indirizzi nell'area ausiliaria vengono ritenute, mentre altre vengono cancellate)
2. Aree con ritenzione: area di ritenzione, area DM, area EM, valori attuali del contatore, flag di completamento del contatore.
3. I dati presenti nella memoria I/O vengono mantenuti quando il bit di ritenzione IOM (A50012) è attivato. Quando il bit di ritenzione IOM (A50012) è attivato e il funzionamento si interrompe a causa di un errore fatale (incluso FALS(007)), il contenuto della memoria I/O verrà mantenuto ma le uscite del Modulo di uscita verranno disattivate.

1-4-3 Modalità di avvio

Per informazioni dettagliate sulla modalità di avvio, consultare il *Manuale dell'operatore* della CPU.

Nota Nel caso delle CPU CJ1, CS1-H, CJ1-H, CJ1M o CS1D, se non è collegata una Console di programmazione, la CPU verrà avviata in modalità RUN. Questa caratteristica differisce dal funzionamento predefinito della CPU CS1, la quale viene avviata in modalità PROGRAM quando non è collegata una Console di programmazione.

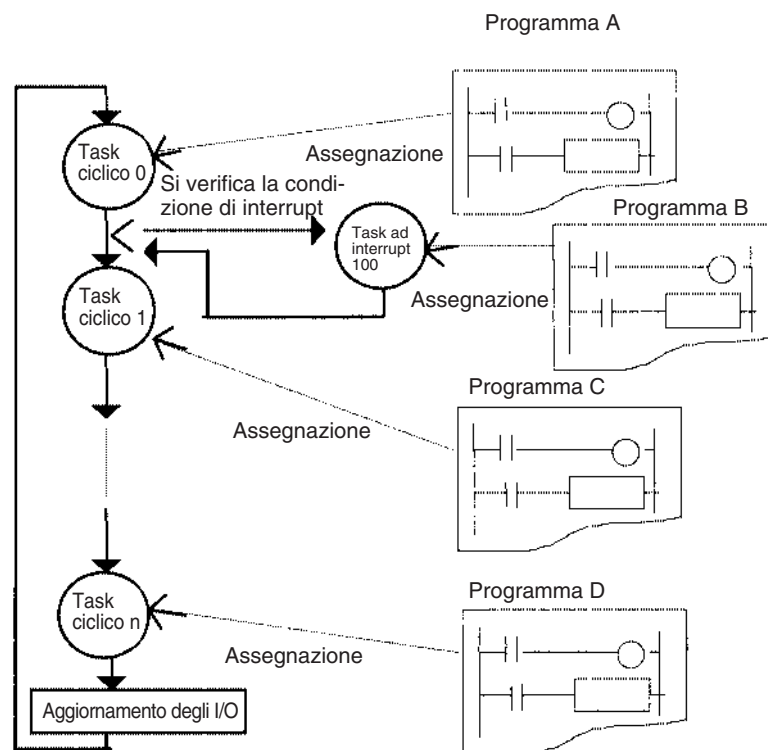
Condizioni	CPU CS1	CPU CJ1, CS1-H, CJ1-H, CJ1M o CS1D
Il PLC è configurato per l'avvio in base alla modalità impostata sulla Console di programmazione, ma non è collegata alcuna Console di programmazione.	Modalità PROGRAM	modalità RUN

1-5 Programmi e task

I task specificano la sequenza e le condizioni di interrupt in base alle quali vengono eseguiti i singoli programmi. Possono essere suddivisi nelle seguenti categorie di massima:

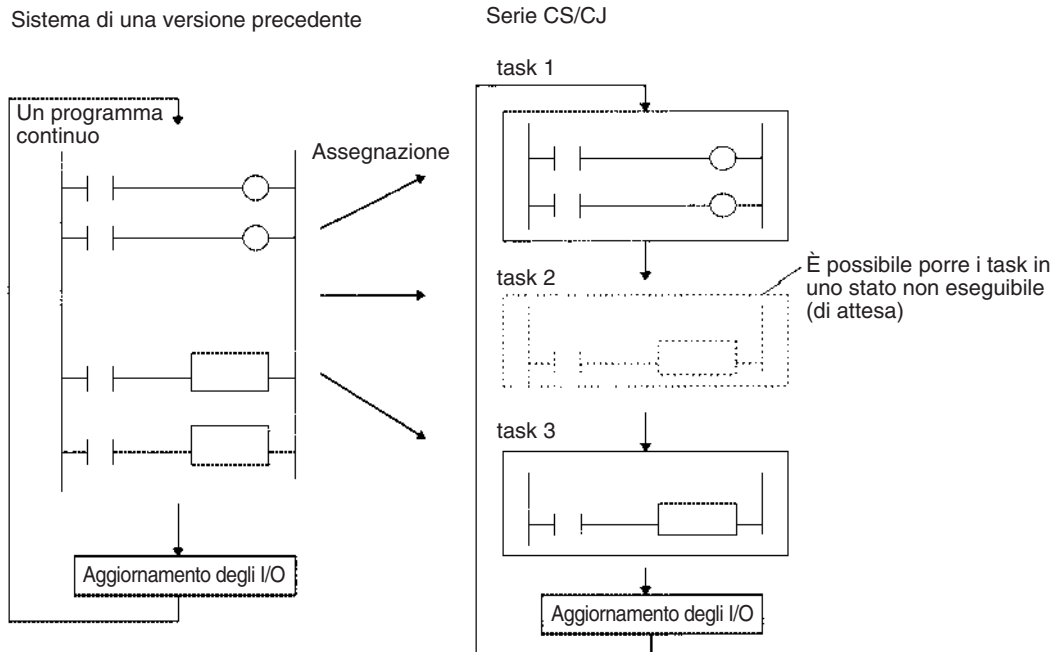
1,2,3...
1. Task eseguiti in sequenza, denominati task ciclici.
2. Task eseguiti in base a condizioni di interrupt, denominati task ad interrupt.

Nota Con le CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt in maniera ciclica, in modo analogo ai task ciclici. Tali task vengono definiti "task ciclici supplementari".

I programmi assegnati a task ciclici vengono eseguiti in sequenza in base al numero di task e gli I/O vengono aggiornati una volta per ciclo al termine dell'esecuzione di tutti i task, o più precisamente di tutti i task con stato eseguibile. Se durante l'elaborazione di un task ciclico si verifica una condizione di interrupt, il task ciclico verrà interrotto e verrà eseguito il programma assegnato al task ad interrupt.

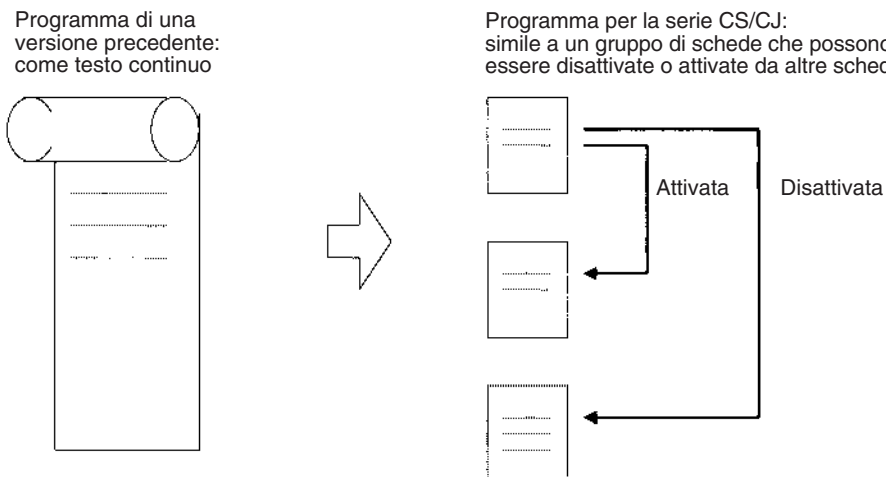
Per ulteriori informazioni sull'aggiornamento degli I/O, consultare il manuale *CS/CJ Series Operation Manual*.

Nell'esempio sopra riportato, la programmazione viene eseguita nel seguente ordine: avvio di A, B, porzione rimanente di A, C, quindi D. Si presume che durante l'esecuzione del programma A si sia verificata la condizione di interrupt per il task ad interrupt 100 e, pertanto, al termine dell'esecuzione del programma B viene eseguito il resto del programma A, a partire dal punto in cui l'esecuzione era stata interrotta.

Nei PLC OMRON meno recenti, un programma continuo è costituito da più parti continue. I programmi assegnati a ciascun task sono programmi singoli che terminano con un'istruzione END, esattamente come per il programma dei PLC meno recenti.

Una caratteristica dei task ciclici consiste nella possibilità di abilitare (stato eseguibile) e disabilitare (stato di attesa) il task tramite istruzioni di controllo. Ne consegue la possibilità di assemblare come task vari componenti di un programma e di eseguire all'occorrenza solo i programmi (task) specifici per il modello di prodotto in uso o la procedura in corso (cambio di step del programma). In questo modo si ottengono prestazioni (tempi del ciclo) nettamente migliori grazie all'esecuzione dei soli programmi richiesti quando necessario.

Un task che è stato eseguito verrà eseguito nei cicli successivi, mentre un task in attesa resterà in attesa nei cicli successivi a meno che la relativa esecuzione non venga richiamata da un altro task.

Nota Diversamente dai programmi della generazione precedente, la cui esecuzione può essere paragonata alla lettura di un testo continuo, l'esecuzione dei task è equiparabile alla lettura di una serie di schede singole.

- Tutte le schede vengono lette secondo un ordine prefissato, a partire da quella avente il numero più basso.
- Tutte le schede vengono contrassegnate come attive o non attive, e queste ultime vengono ignorate. L'attivazione o la disattivazione viene specificata utilizzando istruzioni di controllo dei task.

- Una scheda attivata resterà tale e verrà letta nelle sequenze successive. Una scheda disattivata resterà tale e verrà ignorata nelle sequenze successive, finché non verrà riattivata da un'altra scheda.

1-6 Descrizione dei task

I task possono essere suddivisi nelle seguenti categorie di massima:

1,2,3...

1. Task ciclici (massimo 32)

   Tali task verranno eseguiti una volta per ciclo qualora abbiano stato eseguibile. Se necessario, è possibile disabilitare l'esecuzione dei task ciclici.

2. Task ad interrupt

   Tali task vengono eseguiti quando si verifica una condizione di interrupt, indipendentemente dal fatto che sia in corso l'esecuzione di un task ciclico. I task ad interrupt (vedere note 1 e 2) vengono raggruppati nei seguenti quattro tipi (cinque, se si considerano anche i task ciclici supplementari per le CPU CS1-H, CJ1-H, CJ1M o CS1D):

   a) Task ad interrupt di spegnimento (non supportato dalle CPU CS1D per sistemi a due CPU):
      viene eseguito quando l'alimentazione viene interrotta (massimo 1).

   b) Task ad interrupt programmato (non supportato dalle CPU CS1D per sistemi a due CPU):
      viene eseguito a intervalli prefissati (massimo 2).

   c) Task ad interrupt di I/O (non supportato dalle CPU CS1D o CJ1 per sistemi a due CPU):
      viene eseguito quando viene attivato un ingresso di un Modulo di interrupt di ingresso (massimo 32).

   d) Task ad interrupt esterno (non supportato dalle CPU CJ1 o CS1D per sistemi a due CPU):
      viene eseguito (massimo 256) quando richiesto da un Modulo di I/O speciale, un Modulo CPU bus o una scheda interna (solo serie CS).

   e) Task ciclico supplementare (supportato solo dalle CPU CS1-H, CJ1-H, CJ1M e CS1D)
      Tali task ad interrupt che vengono trattati come task ciclici. I task ciclici supplementari vengono eseguiti una volta per ciclo finché hanno stato eseguibile.

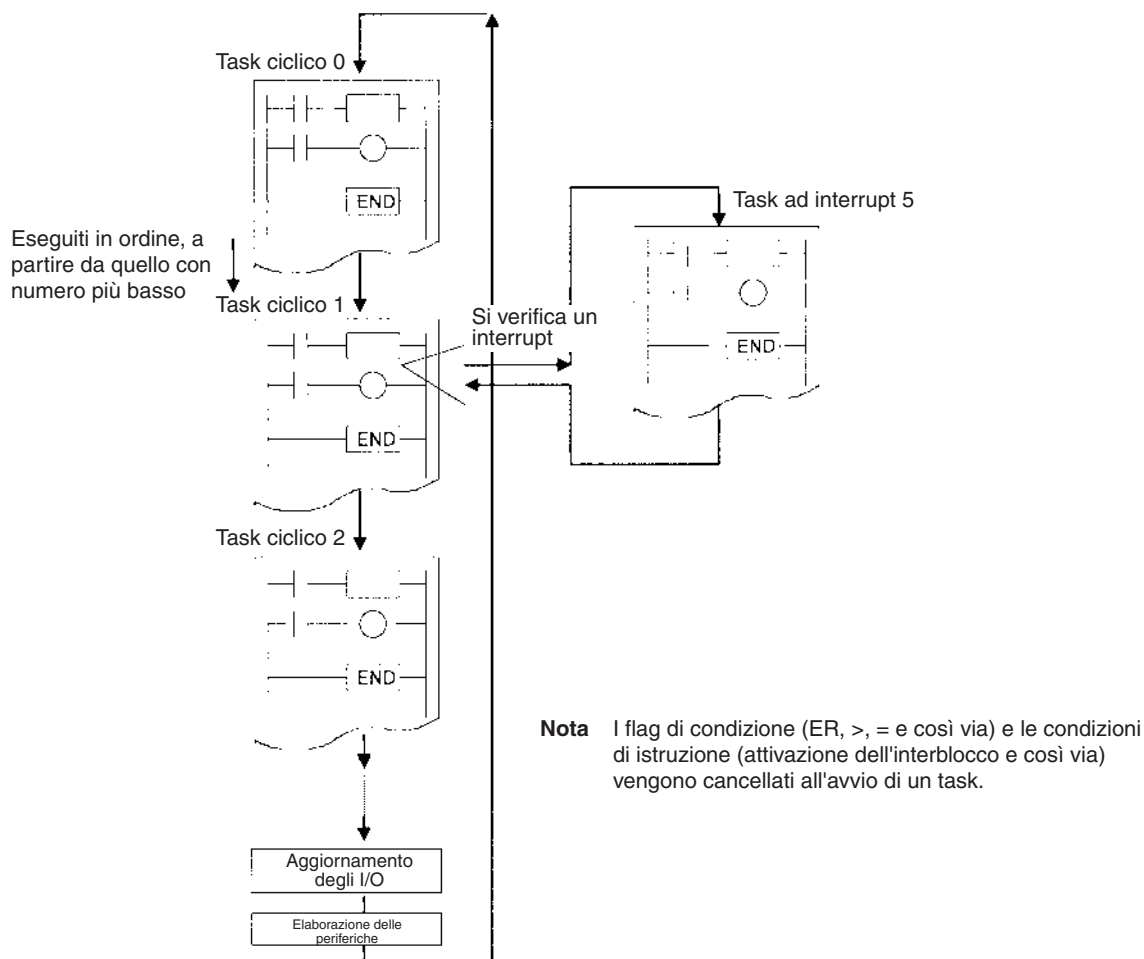
È possibile creare e controllare tramite CX-Programmer un totale di 288 task con 288 programmi. In tale numero sono compresi 32 task ciclici al massimo e 256 task ad interrupt.

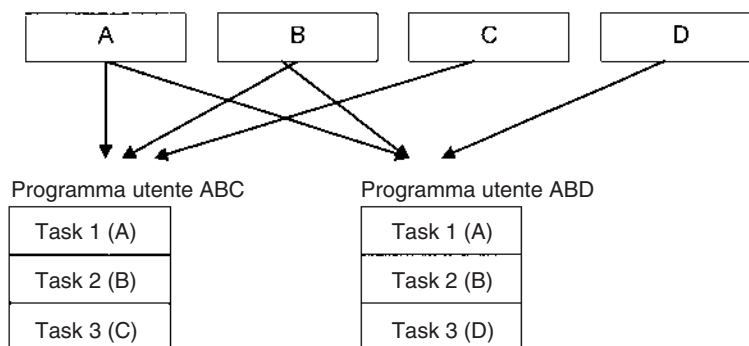
Nota

1. Le CPU CJ1 attualmente non supportano task ad interrupt esterni e task ad interrupt di I/O. Il numero massimo di task per una CPU CJ1 è quindi 35, ovvero 32 task ciclici e 3 task ad interrupt. Pertanto, anche il numero massimo di programmi che è possibile creare e gestire è 35.

2. Le CPU CS1D non supportano alcun task ad interrupt. Tuttavia, è possibile utilizzare task ciclici supplementari.

Ciascun programma viene assegnato a un solo task mediante le impostazioni delle proprietà del programma definite con CX-Programmer.

Nota I flag di condizione (ER, >, = e così via) e le condizioni di istruzione (attivazione dell'interblocco e così via) vengono cancellati all'avvio di un task.

Struttura del programma

È possibile creare programmi di subroutine standard e assegnarli ai task, in base alle esigenze di creazione dei programmi. Ciò significa che è possibile creare moduli di programma (componenti standard) ed eseguire il debug dei singoli task.

Durante la creazione di programmi modulari, è possibile specificare gli indirizzi per mezzo di simboli, in modo da facilitare la standardizzazione.

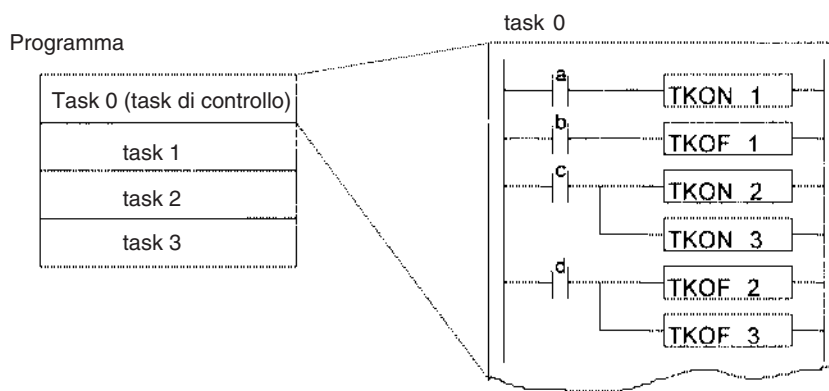
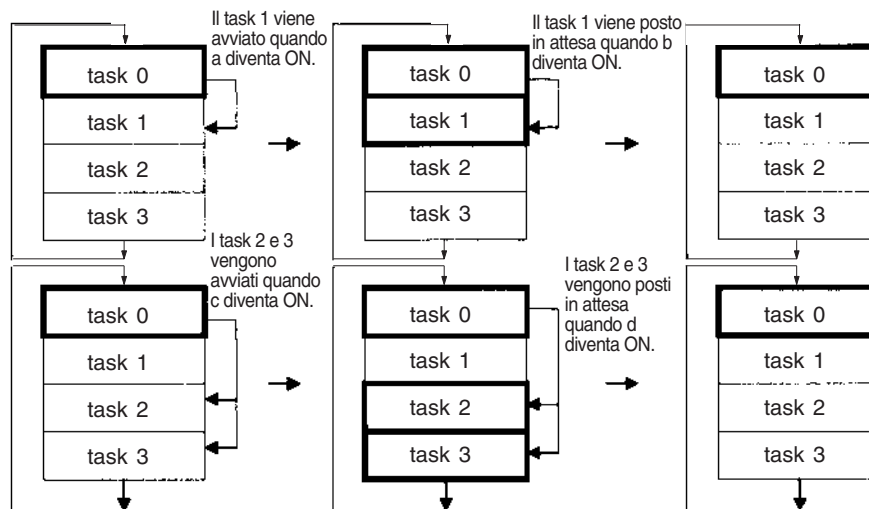
Stato eseguibile e stato di attesa

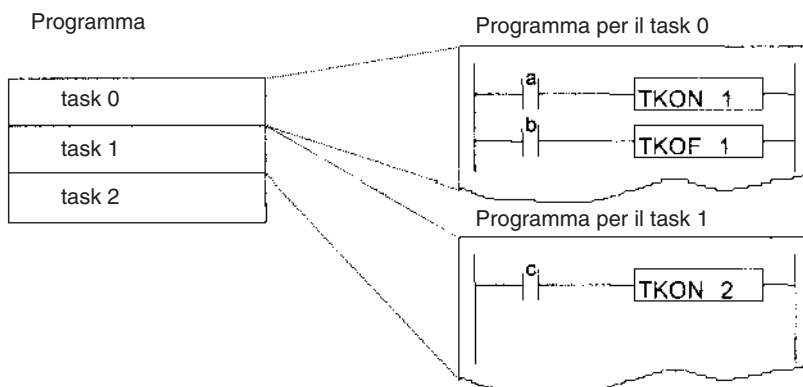
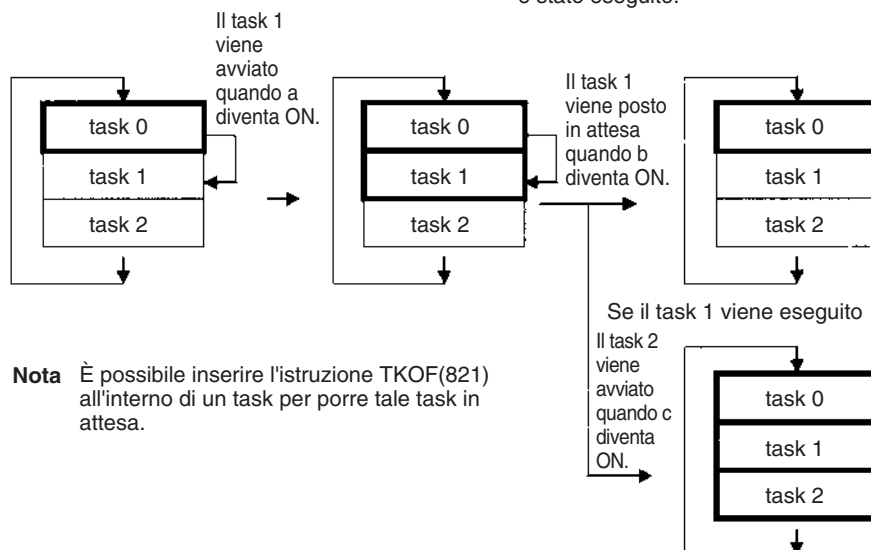
È possibile eseguire le istruzioni TASK ON e TASK OFF (TKON(820) e TKOF(821)) all'interno di un task per impostare lo stato eseguibile o di attesa di un altro task.

Sebbene le istruzioni dei task in stato di attesa non vengano eseguite, il relativo stato degli I/O viene mantenuto. Quando lo stato di un task viene nuovamente impostato come eseguibile, le istruzioni vengono eseguite con lo stato degli I/O mantenuto.

Esempio programmazione con task di controllo

In questo esempio, il task 0 è un task di controllo che viene eseguito per primo all'avvio dell'operazione. È possibile impostare altri task con CX-Programmer (ma non con una Console di programmazione) per l'avvio o meno all'inizio dell'operazione.

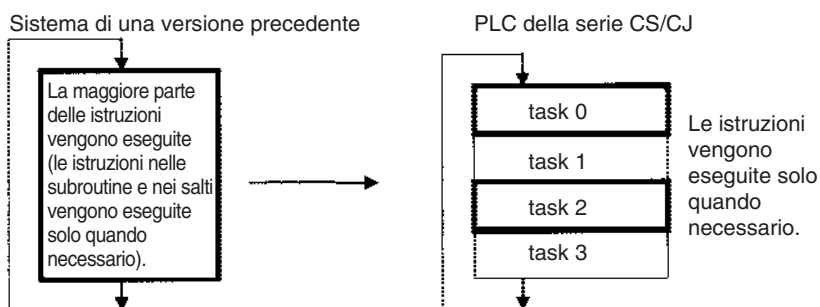
Una volta avviata l'esecuzione del programma, è possibile controllare i task tramite le istruzioni TKON(820) e TKOF(821).

Esempio All'avvio viene eseguito il task 0 (come impostato nelle proprietà del programma tramite CX-Programmer).
Il task 1 è eseguibile quando a diventa ON.
Il task 1 viene posto in attesa quando b diventa ON.
I task 2 e 3 sono eseguibile quando c diventa ON.
I task 2 e 3 vengono posti in attesa quando d diventa ON.

Esempio task controllati da altri task

In questo esempio, ciascun task è controllato da un altro task.

Esempio Per impostazione all'avvio viene eseguito il task 1, indipendentemente da qualsiasi altra condizione.
Il task 1 è eseguibile quando a diventa ON.
Il task 1 viene posto in attesa quando b diventa ON.
Il task 2 è eseguibile quando c diventa ON e il task 1 è stato eseguito.

Nota È possibile inserire l'istruzione TKOF(821) all'interno di un task per porre tale task in attesa.

Tempo di esecuzione dei task

Quando un task è in attesa, le istruzioni contenute in tale task non vengono eseguite e, pertanto, il tempo di esecuzione delle istruzioni disabilitate non viene aggiunto al tempo di ciclo.

Nota Da questo punto di vista, le istruzioni di un task in attesa equivalgono alle istruzioni di una sezione del programma saltata (JMP-JME).

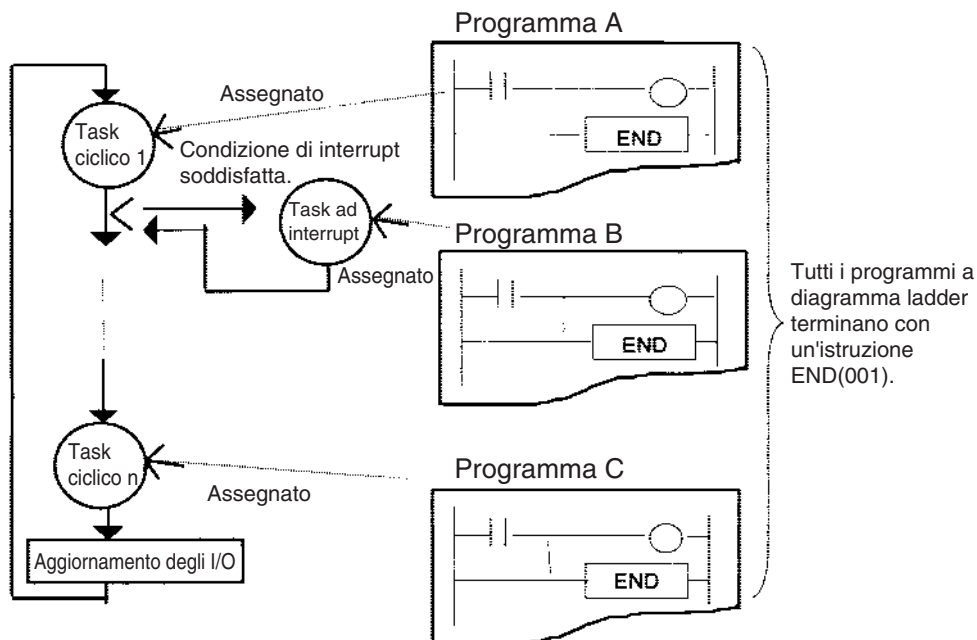
Poiché le istruzioni di un task non eseguito non determinano un aumento del tempo di ciclo, è possibile migliorare in modo considerevole le prestazioni complessive del sistema suddividendo il sistema in un task di controllo globale e in singoli task che vengono eseguiti solo quando necessario.

CAPITOLO 2
Programmazione

In questo capitolo vengono fornite informazioni di base sulla creazione, la verifica e l'immissione di programmi.

2-1 Concetti di base

2-1-1 Programmi e task

I PLC delle serie CS e CJ eseguono programmi a diagramma ladder contenuti in task. Analogamente ai PLC tradizionali, il programma a diagramma ladder all'interno di ciascun task termina con un'istruzione END(001).

I task vengono utilizzati per determinare l'ordine di esecuzione dei programmi a diagramma ladder e le condizioni per l'esecuzione degli interrupt.

In questo capitolo vengono illustrati i concetti di base relativi alla creazione di programmi per i PLC delle serie CS e CJ. Per ulteriori informazioni sui task e sulle relazioni tra task e programmi a diagramma ladder, vedere *CAPITOLO 4 Task.*

Nota Task e dispositivi di programmazione
La gestione dei task con i dispositivi di programmazione è descritta di seguito. Per informazioni più dettagliate, fare riferimento al capitolo *4-4 Operazioni dei dispositivi di programmazione relative ai task,* al manuale *CS/CJ-series Programming Consoles Operation Manual* (W341) e al manuale *CX-Programmer Operation Manual*.

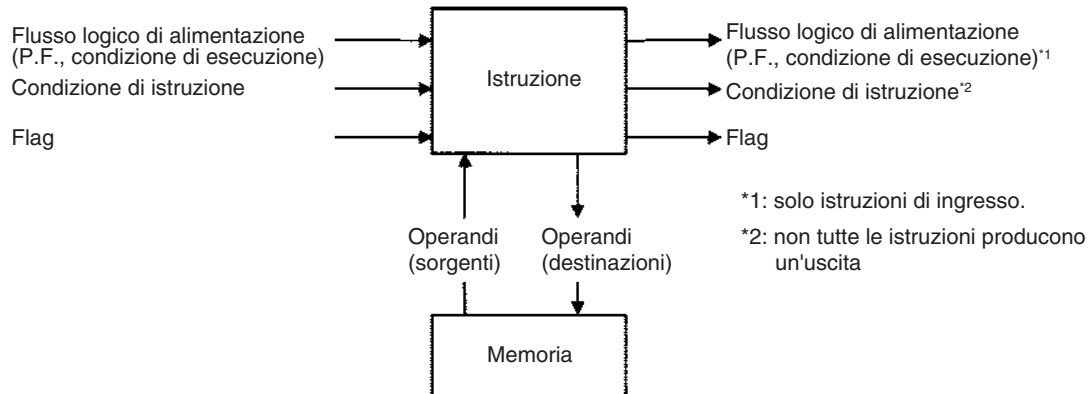
CX-Programmer

CX-Programmer viene utilizzato per specificare i tipi di task e i numeri dei task come attributi per i singoli programmi.

Console di programmazione

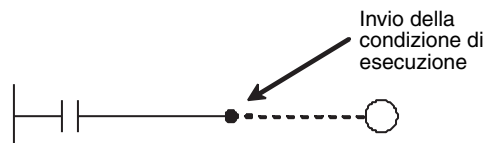
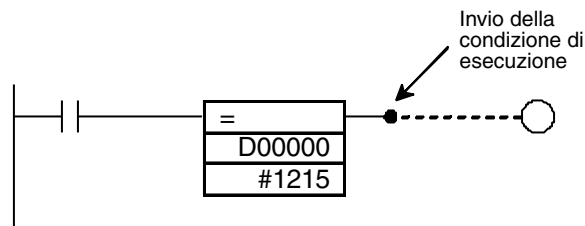
È possibile accedere e modificare i programmi con una Console di programmazione specificando i task ciclici da CT00 a CT31 e i task ad interrupt da IT00 a IT255. Quando l'operazione di cancellazione della memoria viene eseguita mediante una Console di programmazione, è possibile scrivere in un nuovo programma solo il task ciclico 0 (CT00). Per creare i task ciclici da 1 a 31 (da CT01 a CT31), è necessario utilizzare CX-Programmer.

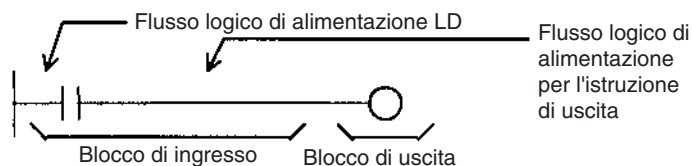
2-1-2 Informazioni di base sulle istruzioni

I programmi sono composti da istruzioni. La struttura concettuale degli ingressi e delle uscite di un'istruzione è illustrata nello schema riportato di seguito.

Flusso logico di alimentazione

Il flusso logico di alimentazione è la condizione di esecuzione utilizzata per controllare l'esecuzione e le istruzioni quando i programmi vengono eseguiti in modo normale. In un programma ladder il flusso logico di alimentazione rappresenta lo stato della condizione di esecuzione.

Istruzioni di ingresso

- Le istruzioni di carico indicano un inizio logico e inviano la condizione di esecuzione.

- Le istruzioni intermedie consentono di immettere il flusso logico di alimentazione come condizione di esecuzione e di inviarlo a un'istruzione intermedia o di uscita.

Istruzioni di uscita

Le istruzioni di uscita eseguono tutte le funzioni utilizzando il flusso logico di alimentazione come condizione di esecuzione.

Condizioni di istruzione

Le condizioni di istruzione sono condizioni speciali relative all'esecuzione complessiva delle istruzioni, e vengono generate dalle istruzioni riportate nella tabella fornita di seguito. Le condizioni di istruzione hanno una priorità più elevata rispetto al flusso logico di alimentazione (P.F., Power Flow) e determinano

se eseguire o meno una determinata istruzione. Un'istruzione può essere resa non eseguibile o assumere un comportamento diverso a seconda delle condizioni di istruzione. Le condizioni di istruzione vengono reimpostate (annullate) all'avvio di ogni task, ovvero vengono reimpostate al cambio di task.

Le istruzioni riportate di seguito vengono utilizzate in coppia per impostare e annullare determinate condizioni di istruzione. È necessario che la coppia di istruzioni appartenga allo stesso task.

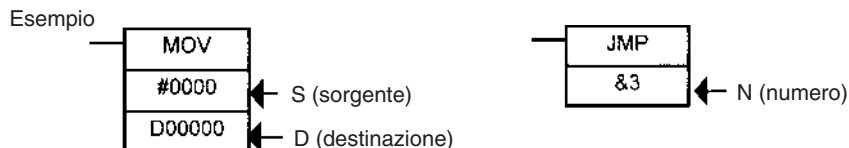
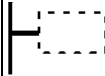
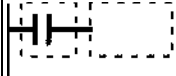
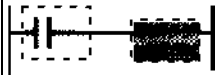
Condizione di istruzione	Descrizione	Istruzione di impostazione	Istruzione di annullamento
Interblocco	Un interblocco consente di disattivare parte del programma. Vengono applicate condizioni speciali quali la disattivazione dei bit di uscita, il ripristino dei temporizzatori e la ritenzione dei contatori.	IL(002)	ILC(003)
Esecuzione di BREAK(514)	Termina un ciclo FOR(512)-NEXT(513) durante l'esecuzione. Impedisce l'esecuzione di tutte le istruzioni fino all'istruzione NEXT(513).	BREAK(514)	NEXT(513)
	Esegue un salto da JMP0(515) a JME0(516).	JMP0(515)	JME0(516)
Esecuzione di programmi a blocchi	Esegue un blocco di programma da BPRG(096) a BEND(801).	BPRG(096)	BEND(801)

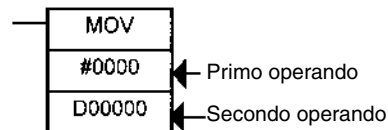
Flag

In questo contesto, un flag corrisponde a un bit che funge da interfaccia tra le istruzioni.

Flag di ingresso	Flag di uscita
• Flag comparativi Flag dei risultati comparativi Lo stato di questi flag viene immesso automaticamente nell'istruzione per tutte le istruzioni di uscita differenziali up/down e le istruzioni DIFU(013) e DIFD(014). • Flag di riporto (CY) Il flag di riporto viene utilizzato come operando non specificato nelle istruzioni di scorrimento dei dati e nelle istruzioni di addizione e sottrazione. • Flag per istruzioni speciali Questi flag includono i flag di autoimpostazione per le istruzioni FPD(269) e i flag di abilitazione della comunicazione in rete.	• Flag comparativi Flag dei risultati comparativi Lo stato di questi flag viene determinato automaticamente dall'istruzione per tutte le istruzioni di uscita differenziali up/down e le istruzioni UP(521) e DOWN(522). • Flag di condizione I flag di condizione includono i flag di attivazione e disattivazione permanente nonché i flag che vengono aggiornati in base ai risultati dell'esecuzione delle istruzioni. Nei programmi utente è possibile specificare tali flag per mezzo di etichette, quali ER, CY, >, =, A1, A0, anziché mediante indirizzi. • Flag per istruzioni speciali Includono i flag delle istruzioni della schedina di memoria e i flag di completamento dell'esecuzione di MSG(046).

Operandi

Gli operandi specificano parametri di istruzione preimpostati (caselle dei diagrammi ladder) che vengono utilizzati per definire le costanti o il contenuto dell'area della memoria I/O. È possibile eseguire un'istruzione immettendo un indirizzo o una costante come operandi. Gli operandi vengono classificati come operandi di origine, di destinazione o di numero.

Tipi di operandi		Simbolo dell'operando	Descrizione	
sorgente	Specifica l'indirizzo dei dati da leggere o una costante.	S	Operando di origine	Operando di origine diverso dai dati di controllo (C)
		C	dati di controllo	Dati compositi di un operando di origine con significati differenti a seconda dello stato dei bit.
Destinazione (risultati)	Specifica l'indirizzo in cui verranno scritti i dati.	D (R)	---	
Numero	Specifica un determinato numero utilizzato nell'istruzione, quale un numero di salto o di subroutine.	N	---	

Nota Gli operandi vengono altresì definiti primo operando, secondo operando e così via, a partire dall'inizio dell'istruzione.

2-1-3 Posizione delle istruzioni e condizioni di esecuzione

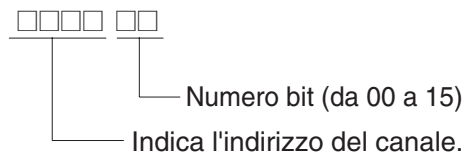
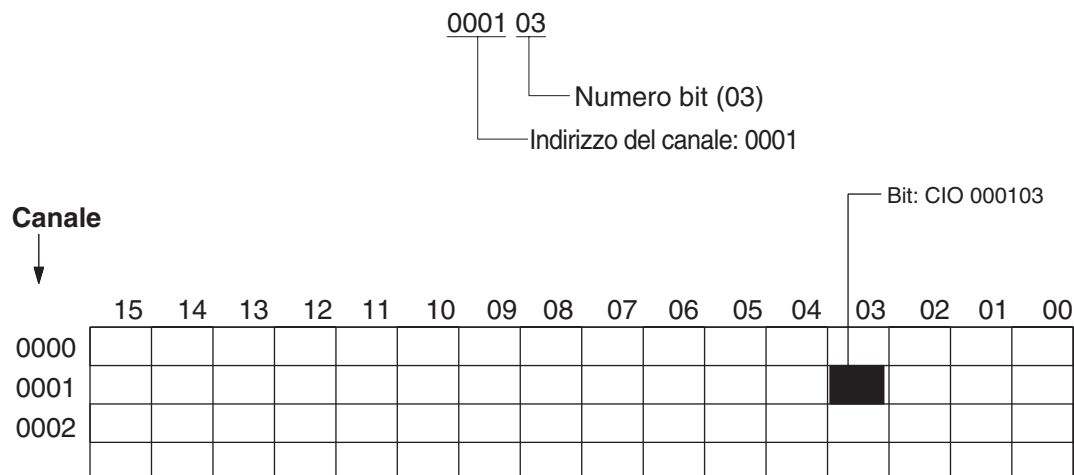
La seguente tabella illustra le possibili posizioni per le istruzioni. Le istruzioni vengono raggruppate in istruzioni che richiedono condizioni di esecuzione e istruzioni che non ne richiedono. Per informazioni dettagliate sulle singole istruzioni, vedere *CAPITOLO 3 Funzioni di istruzione.*

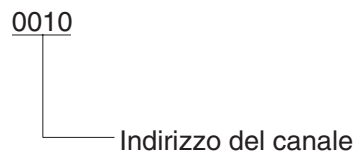
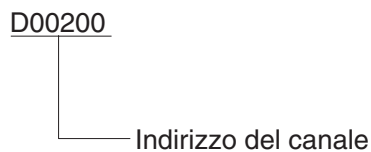
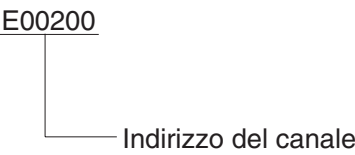
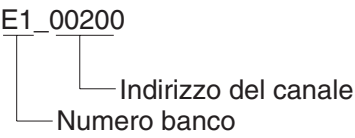
Tipo di istruzione		Possibile posizione	Condizione di esecuzione	Diagramma	Esempi
Istruzioni di ingresso	Inizio logico (istruzioni di carico)	Collegate direttamente alla barra di distribuzione sinistra o posizionate all'inizio di un blocco di istruzioni.	Non richiesta		LD, LD TST(350), LD > (e altre istruzioni di confronto con simboli)
	Istruzioni intermedie	Tra un inizio logico e l'istruzione di uscita.	Richiesta		AND, OR, AND TEST(350), AND > (e altre istruzioni di confronto con simboli ADD), UP(521), DOWN(522), NOT(520) ecc.
Istruzioni di uscita		Collegate direttamente alla barra di distribuzione destra.	Richiesta		La maggior parte delle istruzioni, incluse OUT e MOV(021)
			Non richiesta		END(001), JME(005), FOR(512), ILC(003) ecc.

Nota

1. Esiste un altro gruppo di istruzioni che eseguono una serie di istruzioni mnemoniche sulla base di un singolo ingresso. Tali istruzioni sono definite istruzioni di programmazione a blocchi. Per ulteriori informazioni sui programmi a blocchi, fare riferimento al manuale *CS/CJ Series CPU Units Instruction Reference Manual*.
2. Se un'istruzione che richiede una condizione di esecuzione è collegata direttamente alla barra di distribuzione sinistra senza un'istruzione di inizio logico, durante la verifica del programma da un dispositivo di programmazione (CX-Programmer o Console di programmazione) si verificherà un errore di programma.

2-1-4 Indirizzamento delle aree della memoria I/O

Indirizzi di bit

Esempio l'indirizzo del bit 03 nel canale 0001 dell'area CIO si presenta come illustrato di seguito. In questo manuale tale indirizzo viene identificato come "CIO 000103".

Indirizzi di canale

Esempio l'indirizzo dei bit da 00 a 15 nel canale 0010 dell'area CIO si presenta come illustrato di seguito. In questo manuale tale indirizzo viene identificato come "CIO 0010".

Gli indirizzi delle aree DM ed EM vengono forniti con il prefisso "D" o "E", come riportato di seguito per l'indirizzo D00200.

Esempio l'indirizzo del canale 2000 nel banco corrente della memoria dei dati estesa si presenta nel seguente modo.

L'indirizzo del canale 2000 nel banco 1 della memoria dei dati estesa si presenta nel seguente modo.

2-1-5 Specifica degli operandi

Operando	Descrizione	Notazione	Esempi di applicazione
Specifica degli indirizzi di bit	Il canale e i numeri di bit vengono specificati direttamente per indicare un bit (bit di ingresso) □□□□ □□ Numero bit (da 00 a 15) Indica l'indirizzo del canale. **Nota** Vengono utilizzati gli stessi indirizzi per accedere ai flag di completamento e ai valori attuali del temporizzatore/contatore. Per un flag di task è disponibile un solo indirizzo.	0001 02 Numero bit (02) Numero canale: 0001	0001 02 ─┤├─
Specifica degli indirizzi di canale	Il numero del canale viene specificato direttamente per indicare il canale a 16 bit. □□□□ Indica l'indirizzo del canale.	0003 Numero canale: 0003 D00200 Numero canale: 00200	MOV 0003 D00200

Operando	Descrizione	Notazione	Esempi di applicazione
Specifica degli indirizzi DM/ EM indiretti in modalità binaria	Viene specificato l'offset dall'inizio dell'area. Il contenuto dell'indirizzo verrà trattato come dati in formato binario (da 0000 a 32767) per speci-ficare l'indirizzo del canale nella memoria dei dati (DM) e nella memoria dei dati estesa (EM). Aggiungere il simbolo @ davanti all'indirizzo per specificare un indirizzo indiretto in modalità binaria. @D□□□□□ Contenuto [ ] Da 00000 a 32767 (da 000 a esadecimale a 7FFF esadecimale in binario) D [ ]		
	1) Gli indirizzi da D00000 a D32767 vengono specificati se @D(□□□□□) contiene valori esadecimali da 0000 a 7FFF (da 00000 a 32767).	@D00300 0 1 0 0 Contenuto Binario: 256 Specifica D00256. Aggiungere il simbolo @.	MOV #0001 @00300
	2) Gli indirizzi da E0 _00000 a E0 _32767 del banco 0 nella memoria dei dati estesa (EM) vengono specificati se @D(□□□□□) contiene valori esadecimali da 8000 a FFFF (da 32768 a 65535).	@D00300 8 0 0 1 Contenuto Binario: 32769 Specifica E0 00001.	
	3) Gli indirizzi da E□_00000 a E□_32767 nel banco specificato vengono specificati se @E□_□□□□□ contiene valori esadecimali da 0000 a 7FFF (da 00000 a 32767).	@E1_00200 0 1 0 1 Contenuto Binario: 257 Specifica E1_00257.	MOV #0001 @E1_00200
	4) Gli indirizzi da E(□+1)_00000 a E(□+1)_32767 nel banco successivo al banco specificato □ vengono specificati se @E□_□□□□□ contiene valori esadecimali da 8000 a FFFF (da 32768 a 65535).	@E1_00200 8 0 0 2 Contenuto Binario: 32770 Specifica E2_00002.	
	Nota Quando si specifica un indirizzo indiretto in modalità binaria, considerare la memoria dei dati (DM) e la memoria dei dati estesa (EM) (banchi da 0 a C) come una singola serie di indirizzi. Se il contenuto di un indirizzo con il simbolo @ è superiore a 32767, verrà considerato come indirizzo appartenente alla memoria dei dati estesa (EM) a partire da 00000 nel banco N. 0. **Esempio** Se il canale della memoria dei dati (DM) contiene 32768, verrà specificato l'indirizzo E1_00000 nel banco 0 della memoria dei dati estesa (EM). **Nota** Se il numero di banco della memoria dei dati estesa (EM) viene specificato come "n" e il contenuto del canale è superiore a 32767, l'indirizzo verrà considerato come appartenente alla memoria dei dati estesa (EM) a partire da 00000 nel banco N+1. **Esempio** Se il banco 2 della memoria dei dati estesa (EM) contiene 32768, verrà specificato l'indirizzo E3_00000 nel banco 3 della memoria dei dati estesa (EM).		

Operando	Descrizione	Notazione	Esempi di applicazione
Specifica degli indirizzi DM/ EM indiretti in modalità BCD	Viene specificato l'offset dall'inizio dell'area. Il contenuto dell'indirizzo verrà trattato come dati in formato BCD (da 0000 a 9999) per specificare l'indirizzo del canale nella memoria dei dati (DM) e nella memoria dei dati estesa (EM). Aggiungere un asterisco (*) davanti all'indirizzo per specificare un indirizzo indiretto in modalità BCD. *D□□□□□ Contenuto: Da 00000 a 9999 (BCD) D	*D00200 0 1 0 0 Contenuto Specifica D0100 Aggiungere un asterisco (*)	MOV #0001 *D00200

Operando	Descrizione		Notazione	Esempi di applicazione
Specifica diretta di un registro	Un registro indice (IR) o un registro dati (DR) viene specificato in modo diretto impostando IR□ (□: da 0 a 15) o DR□ (□: da 0 a 15).		IR0 IR1	MOVR 000102 IR0 Memorizza l'indirizzo di memoria del PLC per CIO 0010 in IR0. MOVR 0010 IR1 Memorizza l'indirizzo di memoria del PLC per CIO 0010 in IR1.
Specifica di un indirizzo indiretto mediante un registro	Indirizzo indiretto (nessun offset)	Viene specificato il bit o il canale con l'indirizzo di memoria del PLC contenuto in IR□. Per specificare i bit e i canali per gli operandi di istruzione, specificare ,IR□.	,IR0 ,IR1	LD ,IR0 Carica il bit con l'indirizzo di memoria del PLC contenuto in IR0. MOV #0001 ,IR1 Memorizza #0001 nel canale con l'indirizzo di memoria del PLC contenuto in IR1.
	Offset con costante	Viene specificato il bit o il canale con l'indirizzo di memoria del PLC contenuto in IR□ + o – la costante. Specificare *+/– costante* ,IR□. Gli offset con costante sono compresi tra –2048 e +2047 (decimale). L'offset viene convertito in dato binario durante l'esecuzione dell'istruzione.	+5,IR0 +31,IR1	LD +5 ,IR0 Carica il bit con l'indirizzo di memoria del PLC contenuto in IR0 + 5. MOV #0001 +31 ,IR1 Memorizza #0001 nel canale con l'indirizzo di memoria del PLC contenuto in IR1 + 31
	Offset DR	Viene specificato il bit o il canale con l'indirizzo di memoria del PLC contenuto in IR□ + il contenuto di DR□. Specificare DR□ ,IR□. Il contenuto di DR (registro dati) viene considerato come dato binario con segno. Se il valore binario con segno è negativo, al contenuto di IR□ verrà assegnato un offset negativo.	DR0 ,IR0 DR0 ,IR1	LD DR0 ,IR0 Carica il bit con l'indirizzo di memoria del PLC contenuto in IR0 + il valore in DR0. MOV #0001 DR0 ,IR1 Memorizza #0001 nel canale con l'indirizzo di memoria del PLC contenuto in IR1 + il valore in DR0.
	Incremento automatico	Il contenuto di IR□ viene incrementato di +1 o +2 dopo avere associato il valore a un indirizzo di memoria del PLC. +1: specificare ,IR□+ +2: specificare ,IR□ + +	,IR0 ++ ,IR1 +	LD ,IR0 ++ Incrementa il contenuto di IR0 di un valore pari a 2 dopo il caricamento del bit con l'indirizzo di memoria del PLC contenuto in IR0. MOV #0001 ,IR1 + Incrementa il contenuto di IR1 di un valore pari a 1 dopo la memorizzazione di #0001 nel canale con l'indirizzo di memoria del PLC contenuto in IR1.
	Decremento automatico	Il contenuto di IR□ viene diminuito di –1 o –2 dopo avere associato il valore a un indirizzo di memoria del PLC. –1: specificare ,–IR□ –2: specificare ,– –IR□	,– –IR0 ,–IR1	LD ,– –IR0 Dopo avere diminuito il contenuto di IR0 di un valore pari a 2, viene caricato il bit con l'indirizzo di memoria del PLC contenuto in IR0. MOV #0001 ,–IR1 Dopo avere diminuito il contenuto di IR1 di un valore pari a 1, #0001 viene memorizzato nel canale con l'indirizzo di memoria del PLC contenuto in IR1.

Dati	Operando	Formato dei dati	Simbolo	Intervallo	Esempio di applicazione
Costante a 16 bit	Tutti i dati binari o un intervallo limitato di dati binari	Binario senza segno	#	Da #0000 a #FFFF	---
		Decimale con segno	±	Da –32768 a +32767	---
		Decimale senza segno	& (vedere nota)	Da &0 a &65535	---
	Tutti i dati in formato BCD o in un intervallo limitato di dati	BCD	#	Da #0000 a #9999	---
Costante a 32 bit	Tutti i dati binari o un intervallo limitato di dati binari	Binario senza segno	#	Da #00000000 a #FFFFFFFF	---
		Binario con segno	+	Da –2147483648 a +2147483647	---
		Decimale senza segno	& (vedere nota)	Da &0 a &429467295	---
	Tutti i dati in formato BCD o in un intervallo limitato di dati	BCD	#	Da #00000000 a #99999999	---
Stringa di testo	**Descrizione**		**Simbolo**	**Esempi**	---
	I dati delle stringhe di testo vengono memorizzati in formato ASCII (un byte per carattere, ad eccezione dei caratteri speciali) nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra e dal canale all'estrema destra (il più piccolo) al canale all'estrema sinistra. Il valore esadecimale 00 (codice NUL) viene memorizzato nel byte all'estrema destra dell'ultimo canale, nel caso in cui sia presente un numero dispari di caratteri. 0000 esadecimale (2 codici NUL) viene memorizzato nei byte liberi all'estrema sinistra e all'estrema destra dell'ultimo canale + 1, nel caso in cui sia presente un numero pari di caratteri.		---	'ABCDE' 'A' 'B' 'C' 'D' 'E' NUL = 41 42 43 44 45 00 'ABCD' 'A' 'B' 'C' 'D' NUL NUL = 41 42 43 44 00 00	MOV$ D00100 D00200 D00100 41 42 D00101 43 44 D00102 45 00 ↓ D00200 41 42 D00201 43 44 D00202 45 00
	I caratteri ASCII che è possibile utilizzare in una stringa di testo includono caratteri alfanumerici, caratteri Katakana e simboli, ad eccezione dei caratteri speciali. I caratteri sono illustrati nella tabella riportata di seguito.				

Nota La notazione decimale senza segno viene utilizzata solo in CX-Programmer.

Caratteri ASCII

Bit da 0 a 3		Bit da 4 a 7															
Binario		0000	0001	0010	0011	0100	0101	0110	0111	1000	1001	1010	1011	1100	1101	1110	1111
	Esadecimale	0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
0000	0			Spazio	0	@	P	`	p				ｰ	ﾀ	ﾐ		
0001	1			!	1	A	Q	a	q			｡	ｱ	ﾁ	ﾑ		
0010	2			"	2	B	R	b	r			｢	ｲ	ﾂ	ﾒ		
0011	3			#	3	C	S	c	s			｣	ｳ	ﾃ	ﾓ		
0100	4			$	4	D	T	d	t			､	ｴ	ﾄ	ﾔ		
0101	5			%	5	E	U	e	u			･	ｵ	ﾅ	ﾕ		
0110	6			&	6	F	V	f	v			ｦ	ｶ	ﾆ	ﾖ		
0111	7			'	7	G	W	g	w			ｧ	ｷ	ﾇ	ﾗ		
1000	8			(	8	H	X	h	x			ｨ	ｸ	ﾈ	ﾘ		
1001	9			)	9	I	Y	i	y			ｩ	ｹ	ﾉ	ﾙ		
1010	A			*	:	J	Z	j	z			ｪ	ｺ	ﾊ	ﾚ		
1011	B			+	;	K	[	k	{			ｫ	ｻ	ﾋ	ﾛ		
1100	C			,	<	L	¥	l	|			ｬ	ｼ	ﾌ	ﾜ		
1101	D			-	=	M	]	m	}			ｭ	ｽ	ﾍ	ﾝ		
1110	E			.	>	N	^	n	~			ｮ	ｾ	ﾎ	ﾞ		
1111	F			/	?	O	_	o				ｯ	ｿ	ﾏ	ﾟ		

2-1-6 Formati dei dati

Nella tabella riportata di seguito sono illustrati i formati dei dati che i PLC delle serie CS e CJ sono in grado di gestire.

Tipo di dati	Formato dei dati	Decimale	Esadecimale a 4 cifre
Binario senza segno	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Binario 2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^{9} 2^{8} 2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0} Decimale 3276816384 8192 4092 2048 1024 512 256 128 64 12 16 8 4 2 1 Esadecimale 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0}	da 0 a 65535	da 0000 a FFFF
Binario con segno	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Binario 2^{15} 2^{14} 2^{13} 2^{12} 2^{11} 2^{10} 2^{9} 2^{8} 2^{7} 2^{6} 2^{5} 2^{4} 2^{3} 2^{2} 2^{1} 2^{0} Decimale 3276816384 8192 4092 2048 1024 512 256 128 64 12 16 8 4 2 1 Esadecimale 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} Bit di segno: 0: positivo, 1: negativo	da –32768 a +32767	da 8000 a 7FFF
BCD (decimale codificato in binario)	15 14 13 12 11 10 9 8 7 6 5 4 3 2 1 0 Binario 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} 2^{3} 2^{2} 2^{1} 2^{0} Decimale da 0 a 9 da 0 a 9 da 0 a 9 da 0 a 9	da 0 a 9999	da 0000 a 9999

Tipo di dati	Formato dei dati	Decimale	Esadecimale a 4 cifre
Decimale a virgola mobile in singola precisione	31 30 29 23 22 21 20 19 18 17 3 2 1 0 Segno della mantissa — Esponente — Mantissa Binario Valore = $(-1)^{segno}$ x 1.[Mantissa] x $2^{Esponente}$ Segno (bit 31): 1: negativo o 0: positivo Mantissa: I 23 bit da 00 a 22 contengono la mantissa, cioè la parte del numero dopo la virgola decimale 1,□□□......, in formato binario. Esponente: Gli 8 bit da 23 a 30 contengono l'esponente espresso in formato binario come 127 più N in 2^n. **Nota** Questo formato dei dati a virgola mobile in singola precisione è conforme agli standard IEEE754 e viene utilizzato solo con istruzioni che convertono o utilizzano numeri a virgola mobile. Questo formato può essere utilizzato per eseguire impostazioni o monitoraggi dalla pagina di modifica e monitoraggio della memoria I/O di CX-Programmer, ma non è supportato dalle Console di programmazione. Gli utenti non devono necessariamente conoscere i dettagli relativi a questo formato, eccetto per il fatto che occupa 2 canali.	---	---
Decimale a virgola mobile in doppia precisione	63 62 61 52 51 50 49 48 47 46 3 2 1 0 Segno della mantissa — Esponente — Mantissa Binario Valore = $(-1)^{Segno}$ x 1.[Mantissa] x $2^{Esponente}$ Segno (bit 63): 1: negativo o 0: positivo Mantissa: I 52 bit da 00 a 51 contengono la mantissa, cioè la parte del numero dopo la virgola decimale 1,□□□......, in formato binario. Esponente: Gli 11 bit da 52 a 62 contengono l'esponente espresso in formato binario come 1023 più N in 2^n. **Nota** Questo formato dei dati a virgola mobile in doppia precisione è conforme agli standard IEEE754 e viene utilizzato solo con istruzioni che convertono o utilizzano numeri a virgola mobile. Questo formato può essere utilizzato per eseguire impostazioni o monitoraggi dalla pagina di modifica e monitoraggio della memoria I/O di CX-Programmer, ma non è supportato dalle Console di programmazione. Gli utenti non devono necessariamente conoscere i dettagli relativi a questo formato, eccetto per il fatto che occupa 4 canali.	---	---

Dati binari con segno

Nei dati binari con segno il bit all'estrema sinistra indica il segno dei dati binari a 16 bit. Il valore è espresso in formato esadecimale a 4 cifre.

Numeri positivi. Un valore è positivo o pari a 0 se il bit all'estrema sinistra è 0 (OFF). Nel formato esadecimale a 4 cifre, tale valore viene espresso come valore compreso tra 0000 e 7FFF esadecimale.

Numeri negativi. Un valore è negativo se il bit all'estrema sinistra è 1 (ON). Nel formato esadecimale a 4 cifre, tale valore viene espresso come valore compreso tra 8000 e FFFF esadecimale. Il valore assoluto del valore negativo (decimale) viene espresso come complemento a 2.

Esempio Per indicare –19 in formato decimale come valore binario con segno, sottrarre il valore esadecimale 0013 (valore assoluto di 19) dal valore esadecimale FFFF e aggiungere 0001 esadecimale in modo da ottenere FFED esadecimale.

	F	F	F	F
	1111	1111	1111	1111
Numero effettivo	0	0	1	3
–)	0000	0000	0001	0011
	F	F	E	C
	1111	1111	1110	1100
	0	0	0	1
+)	0000	0000	0000	0001
Complemento a 2	F	F	E	D
	1111	1111	1110	1101

Complementi

In genere il complemento in base x indica un numero ottenuto dalla sottrazione di tutte le cifre di un determinato numero da x – 1 e dall'aggiunta di 1 alla cifra all'estrema sinistra. Ad esempio, il complemento a 10 di 7556 è 9999 – 7556 + 1 = 2444. Il complemento viene utilizzato per esprimere la sottrazione e altre funzioni, quali l'addizione.

Esempio 8954 – 7556 = 1398, 8954 + (complemento a 10 di 7556) = 8954 + 2444 = 11398. Se si ignora il bit all'estrema sinistra, il risultato della sottrazione sarà pari a 1398.

Complementi a 2

Un complemento a 2 è un complemento in base 2. In tal caso, si sottraggono tutte le cifre da 1 (2 – 1 = 1) e si aggiunge 1.

Ad esempio, Il complemento a 2 del numero binario 1101 è 1111 (F esadecimale) – 1101 (D esadecimale) + 1 (1 esadecimale) = 0011 (3 esadecimale). Di seguito questo valore viene espresso in formato esadecimale a 4 cifre.

Il complemento a 2 b esadecimale di a esadecimale è FFFF esadecimale – a esadecimale + 0001 esadecimale = b esadecimale. Per determinare il complemento a 2 b esadecimale di "a esadecimale", calcolare b esadecimale = 10000 esadecimale – a esadecimale.

Ad esempio, per determinare il complemento a 2 del valore esadecimale 3039, calcolare 10000 esadecimale – 3039 esadecimale = CFC7 esadecimale.

Analogamente, per derivare il valore a esadecimale dal complemento a 2 b esadecimale, calcolare a esadecimale = 10000 esadecimale – b esadecimale.

Esempio Per derivare il valore reale dal complemento a 2 CFC7 esadecimale, calcolare 10000 esadecimale – CFC7 esadecimale = 3039 esadecimale.

Per le serie CS e CJ sono previste le istruzioni NEG(160) (2'S COMPLEMENT) e NEGL(161) (DOUBLE 2'S COMPLEMENT), che è possibile utilizzare per derivare il complemento a 2 dal numero reale o per derivare il numero reale dal complemento a 2.

Dati BCD con segno

I dati BCD con segno rappresentano un formato di dati speciale utilizzato per esprimere i numeri negativi in formato BCD. Sebbene tale formato venga utilizzato nelle applicazioni, non è definito in modo esplicito e dipende dall'applicazione specifica. I PLC delle serie CS e CJ supportano le seguenti istruzioni

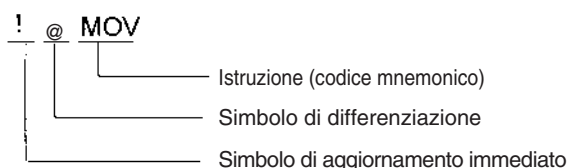
per la conversione dei formati di dati: SIGNED BCD-TO-BINARY: BINS(470), DOUBLE SIGNED BCD-TO-BINARY: BISL(472), SIGNED BINARY-TO-BCD: BCDS(471) e DOUBLE SIGNED BINARY-TO-BCD: BDSL(473). Per ulteriori informazioni, fare riferimento al manuale *CS/CJ-series Programmable Controllers Instructions Reference Manual (W340)*.

Decimale	Esadecimale	Binario	BCD	
0	0	0000		0000
1	1	0001		0001
2	2	0010		0010
3	3	0011		0011
4	4	0100		0100
5	5	0101		0101
6	6	0110		0110
7	7	0111		0111
8	8	1000		1000
9	9	1001		1001
10	A	1010	0001	0000
11	B	1011	0001	0001
12	C	1100	0001	0010
13	D	1101	0001	0011
14	E	1110	0001	0100
15	F	1111	0001	0101
16	10	10000	0001	0110

Decimale	Binario senza segno (esadecimale a 4 cifre)	Binario con segno (esadecimale a 4 cifre)
+65,535	FFFF	Non può essere espresso.
+65534	FFFE	
. . .	. . .	
+32,769	8001	
+32,768	8000	
+32,767	7FFF	7FFF
+32,766	7FFE	7FFE
. . .	. . .	
+2	0002	0002
+1	0001	0001
0	0000	0000
–1	Non può essere espresso.	FFFF
–2		FFFE
. . .		
–32,767		8001
–32,768		8000

2-1-7 Variazioni di istruzione

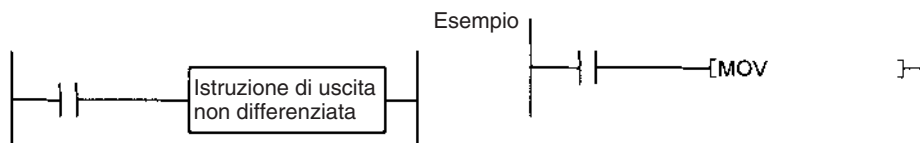
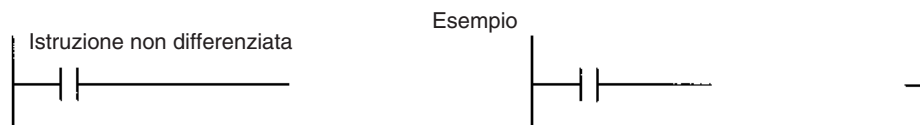
Per differenziare le condizioni di esecuzione e aggiornare i dati quando l'istruzione viene eseguita (aggiornamento immediato), sono disponibili le variazioni di istruzione riportate di seguito.

Variazione		Simbolo	Descrizione
Differenziazione	ON	@	L'istruzione viene differenziata quando la condizione di esecuzione viene attivata.
	OFF	%	L'istruzione viene differenziata quando la condizione di esecuzione viene disattivata.
Aggiornamento immediato		!	I dati nell'area degli I/O specificati dagli operandi o i canali del Modulo di I/O speciale vengono aggiornati all'esecuzione dell'istruzione. L'aggiornamento immediato non è supportato dalle CPU CS1D per sistemi a due CPU.

2-1-8 Condizioni di esecuzione

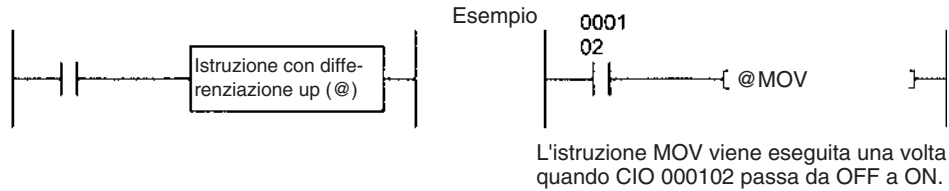
I PLC delle serie CS e CJ utilizzano i seguenti tipi di istruzioni di base e speciali:

- Istruzioni non differenziate eseguite a ogni ciclo
- Istruzioni differenziate eseguite una sola volta

Istruzioni non differenziate

Le istruzioni di uscita che richiedono condizioni di esecuzione vengono eseguite una volta per ciclo quando la condizione di esecuzione è valida (ON o OFF).

Le istruzioni di ingresso per la creazione di inizi logici e le istruzioni intermedie leggono lo stato dei bit, effettuano confronti, verificano i bit o eseguono altri tipi di elaborazione ad ogni ciclo. Se lo stato dei risultati è ON, il flusso logico di alimentazione viene inviato, ovvero viene attivata la condizione di esecuzione.

Istruzioni a ingresso differenziato

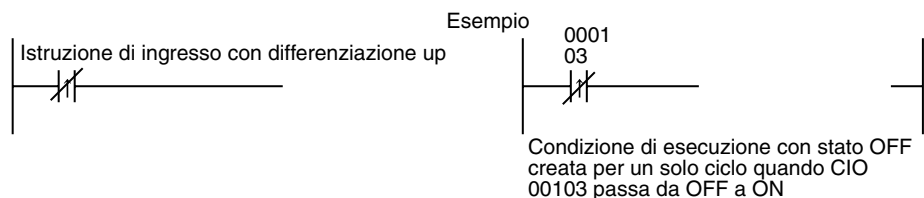
Istruzioni con differenziazione up (istruzione preceduta da @)

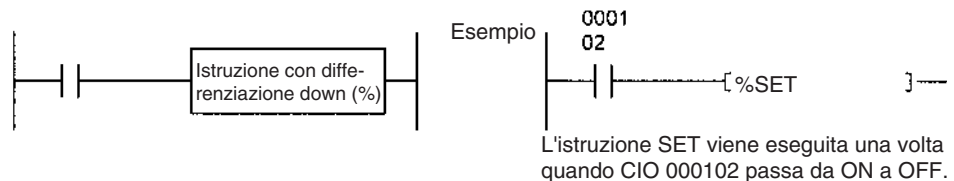
- **Istruzioni di uscita**. L'istruzione viene eseguita solo durante il ciclo in cui la condizione di esecuzione viene attivata (OFF → ON) e non viene eseguita nei cicli successivi.

L'istruzione MOV viene eseguita una volta quando CIO 000102 passa da OFF a ON.

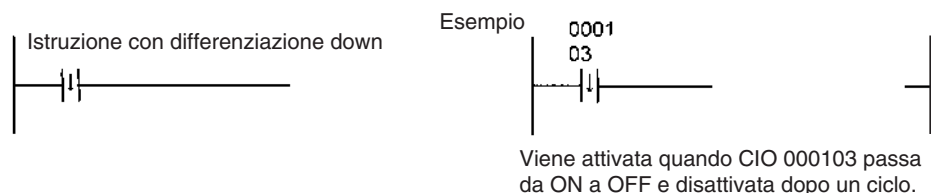
- **Istruzioni di ingresso (inizi logici e istruzioni intermedie)**. L'istruzione legge lo stato dei bit, effettua confronti, verifica i test o esegue altri tipi di elaborazione a ogni ciclo e invia una condizione di esecuzione con stato ON (flusso logico di alimentazione) quando lo stato dei risultati passa da OFF a ON. La condizione di esecuzione viene disattivata al ciclo successivo.

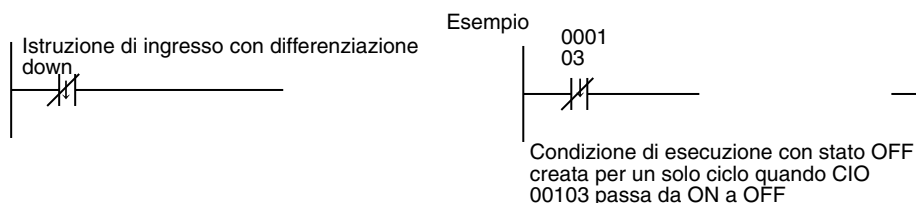
La condizione di esecuzione viene attivata per un solo ciclo quando CIO 000103 passa da OFF a ON.

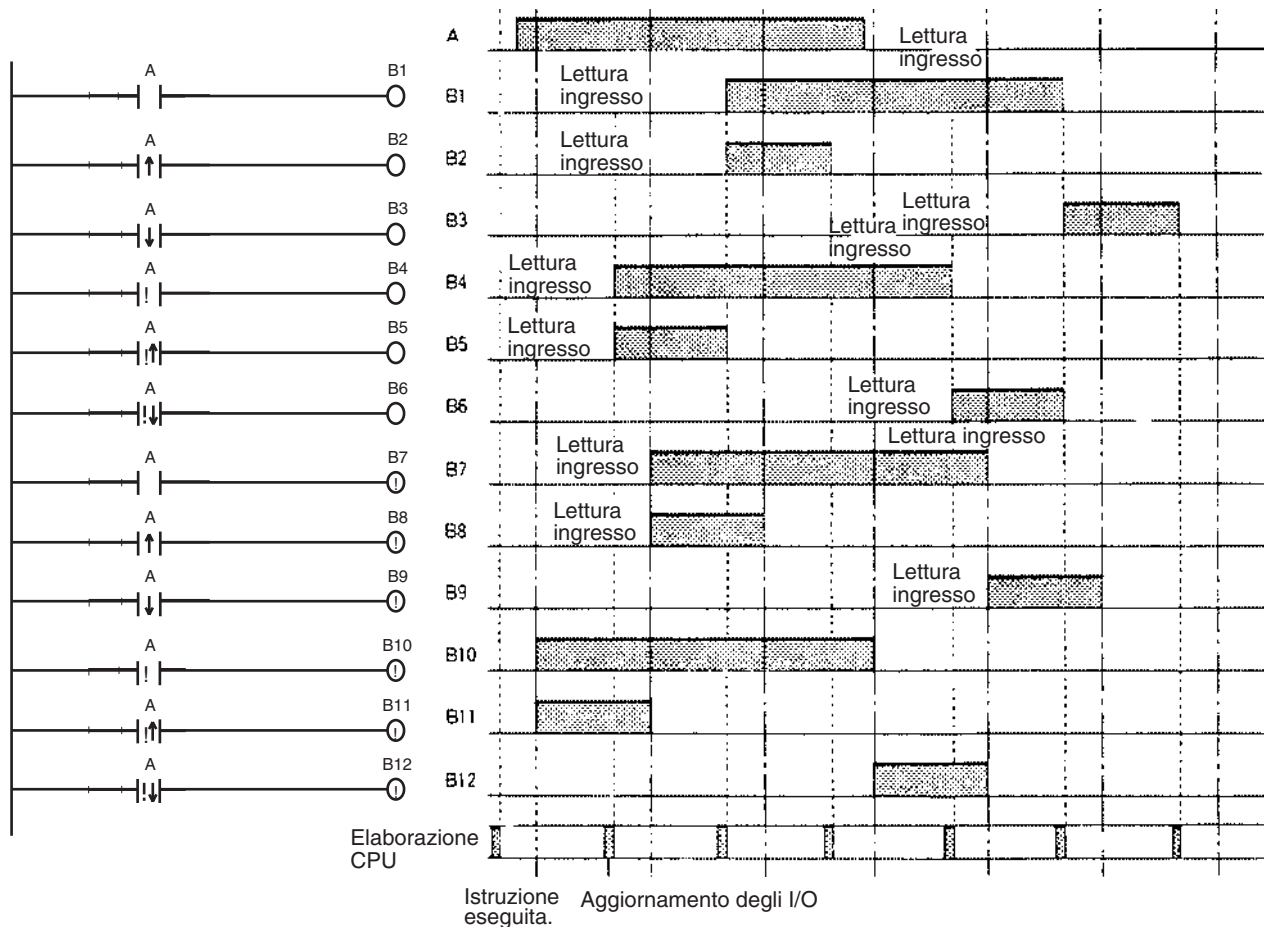
- **Istruzioni di ingresso (inizi logici e istruzioni intermedie)**. L'istruzione legge lo stato dei bit, effettua confronti, verifica i test o esegue altri tipi di elaborazione a ogni ciclo e invia una condizione di esecuzione con stato OFF (interruzione del flusso logico di alimentazione) quando lo stato dei risultati passa da OFF a ON. La condizione di esecuzione viene attivata al ciclo successivo.

Condizione di esecuzione con stato OFF creata per un solo ciclo quando CIO 00103 passa da OFF a ON

Istruzioni con differenziazione down (istruzione preceduta da %)

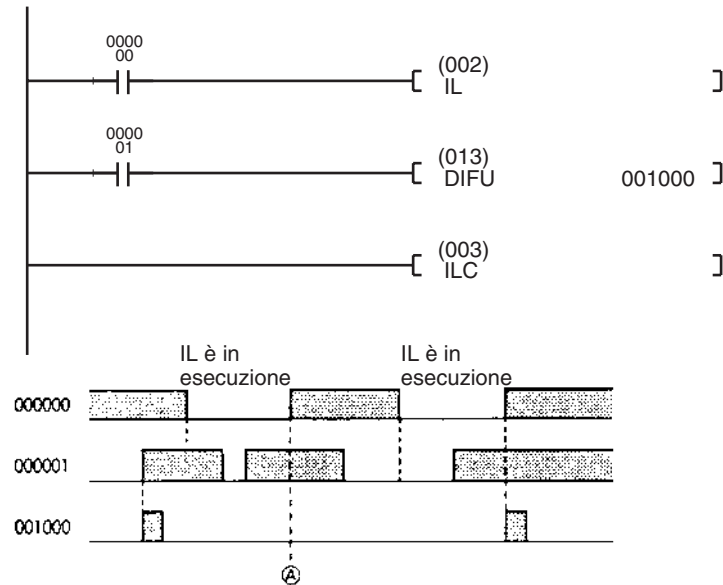
- **Istruzioni di uscita**. L'istruzione viene eseguita solo durante il ciclo in cui la condizione di esecuzione viene disattivata (ON → OFF) e non viene eseguita nei cicli successivi.

L'istruzione SET viene eseguita una volta quando CIO 000102 passa da ON a OFF.

- **Istruzioni di ingresso (inizi logici e istruzioni intermedie)**. L'istruzione legge lo stato dei bit, effettua confronti, verifica i test o esegue altri tipi di elaborazione a ogni ciclo e invia la condizione di esecuzione (flusso logico di alimentazione) quando lo stato dei risultati passa da ON a OFF. La condizione di esecuzione viene disattivata al ciclo successivo.

Nota A differenza delle istruzioni con differenziazione up, la variazione di differenziazione down (%) può essere aggiunta solo alle istruzioni LD, AND, OR, SET e RSET. Per eseguire la differenziazione down con altre istruzioni, combinare tali istruzioni con un'istruzione DIFD o DOWN. È possibile aggiungere NOT alle istruzioni solo quando si utilizza una CPU CS1-H, CJ1-H, CJ1M o CS1D.

- **Istruzioni di ingresso (inizi logici e istruzioni intermedie)**. L'istruzione legge lo stato dei bit, effettua confronti, verifica i test o esegue altri tipi di elaborazione a ogni ciclo e invia una condizione di esecuzione con stato OFF (interruzione del flusso logico di alimentazione) quando lo stato dei risultati passa da ON a OFF. La condizione di esecuzione viene attivata al ciclo successivo.

2-1-9 Tempistica delle istruzioni di I/O

Il diagramma riportato di seguito illustra la differente tempistica operativa di singole istruzioni in un programma composto esclusivamente da istruzioni LD e OUT.

Istruzioni differenziate

- Un'istruzione differenziata dispone di un flag interno che indica se il valore precedente è impostato su ON o OFF. All'avvio del funzionamento, i flag di valore precedente relativi all'istruzione con differenziazione up (istruzioni DIFU e @) vengono impostati su ON e i flag di valore precedente relativi alle istruzioni con differenziazione down (istruzioni DIFD e %) vengono impostati su OFF. Ciò impedisce l'attivazione imprevista di uscite differenziali all'avvio del funzionamento.
- Un'istruzione con differenziazione up (istruzione DIFU o @) verrà attivata solo quando lo stato della condizione di esecuzione è ON e il flag di valore precedente è OFF.

- **Utilizzo negli interblocchi (istruzioni IL - ILC)**
  Nell'esempio riportato di seguito, il flag di valore precedente per l'istruzione differenziata mantiene il valore di interblocco precedente e non attiverà un'uscita differenziale nel punto A in quanto il valore non viene aggiornato finché l'interblocco è in funzione.

- **Utilizzo nei salti (istruzioni JMP - JME).** Analogamente agli interblocchi, il flag di valore precedente per un'istruzione differenziata non viene modificato quando l'istruzione viene saltata, vale a dire viene mantenuto il valore precedente. Le istruzioni con differenziazione up e down genereranno la condizione di esecuzione solo dopo che lo stato dell'ingresso viene modificato rispetto allo stato indicato dal flag di valore precedente.

  Nota a) Non utilizzare il flag di attivazione permanente o il flag A20011 (flag del primo ciclo) come bit di ingresso per un'istruzione con differenziazione up. L'istruzione non verrà mai eseguita.

  b) Non utilizzare il flag di disattivazione permanente come bit di ingresso per un'istruzione con differenziazione down. L'istruzione non verrà mai eseguita.

2-1-10 Frequenza di aggiornamento

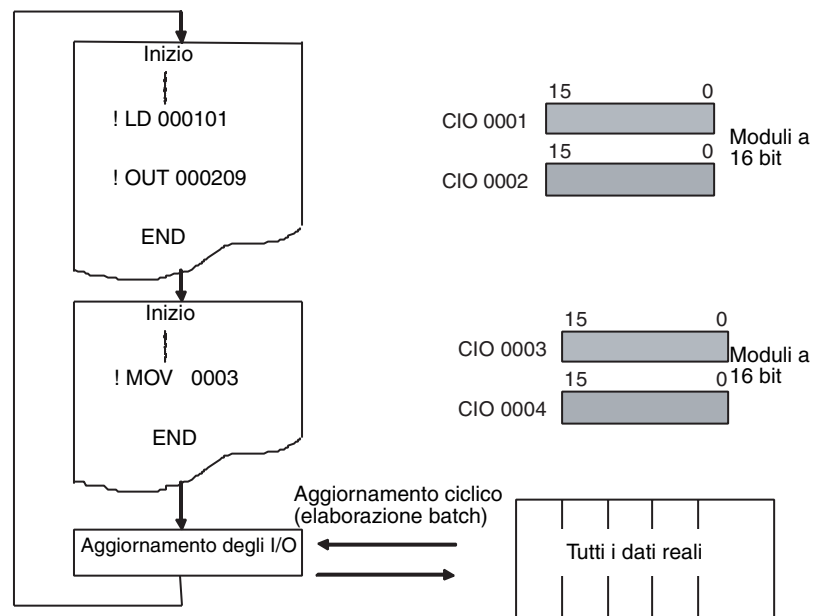
Per aggiornare gli I/O esterni vengono utilizzati i seguenti metodi:

- Aggiornamento ciclico
- Aggiornamento immediato (istruzione preceduta da !, istruzione IORF)

Per ulteriori informazioni sull'aggiornamento degli I/O, fare riferimento al capitolo dedicato al funzionamento della CPU del manuale *CS/CJ Series Operation Manual*.

Aggiornamento ciclico

Ogni programma assegnato a un task ciclico pronto o a un task in cui viene soddisfatta la condizione di interrupt verrà eseguito a partire dall'indirizzo di inizio del programma e l'esecuzione continuerà fino all'istruzione END(001). Una volta eseguiti tutti i task ciclici pronti o i task in cui viene soddisfatta la condizione di interrupt, l'aggiornamento ciclico consentirà di aggiornare contemporaneamente tutti i punti di I/O.

Nota I programmi possono essere eseguiti in più task. In tal caso, gli I/O verranno aggiornati dopo l'istruzione finale END(001) del programma assegnato al numero di task più alto (tra tutti i task ciclici pronti) e non dopo l'istruzione END(001) dei programmi assegnati ad altri task ciclici.

Se occorre procedere all'aggiornamento degli I/O in altri task, eseguire un'istruzione IORF per tutti i canali necessari prima dell'istruzione END(001).

Aggiornamento immediato

Istruzioni con variazione di aggiornamento (!)

Se si specifica un bit di I/O reale come operando gli I/O verranno aggiornati durante l'esecuzione di un'istruzione, come illustrato di seguito.

Moduli	Dati aggiornati
Moduli di I/O di base C200H (solo serie CS)	Gli I/O verranno aggiornati per i 16 bit contenenti il bit specificato.
Moduli di I/O di base CJ	

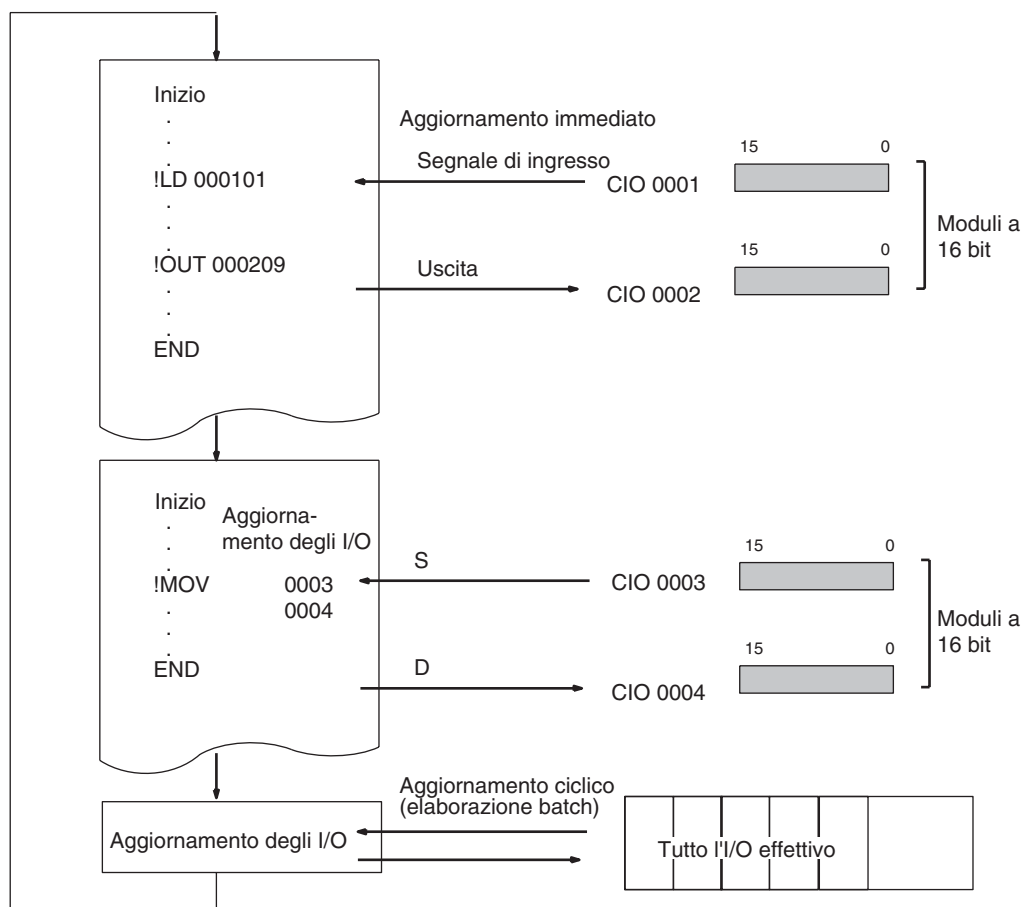
- Quando per un'istruzione si specifica un operando di canale, i punti di I/O vengono aggiornati per i 16 bit specificati.
- Gli ingressi vengono aggiornati per l'operando di ingresso o di origine immediatamente prima dell'esecuzione di un'istruzione.
- Le uscite vengono aggiornate per gli operandi di uscita o di destinazione (D) immediatamente dopo l'esecuzione di un'istruzione.

Aggiungere un punto esclamativo (!) (opzione di aggiornamento immediato) prima dell'istruzione.

Nota Le CPU CS1D per sistemi a due CPU non supportano l'aggiornamento immediato, tuttavia supportano l'aggiornamento tramite istruzioni IORF(097) e DLNK(226).

Aggiornamento dei Moduli tramite istruzione I/O REFRESH

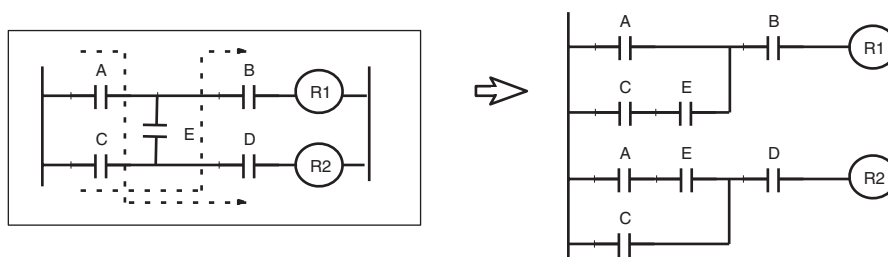
Posizione	Sistema CPU o sistema di I/O di espansione (esclusi sistemi slave SYSMAC BUS)		
Moduli	Moduli di I/O di base	Moduli di I/O di base serie CS/CJ	Aggiornati
		Modulo di I/O di base C200H (vedere nota)	Aggiornati
		Moduli di I/O ad alta densità C200H gruppo 2 (vedere nota)	Non aggiornati
	Moduli di I/O speciale		Non aggiornati

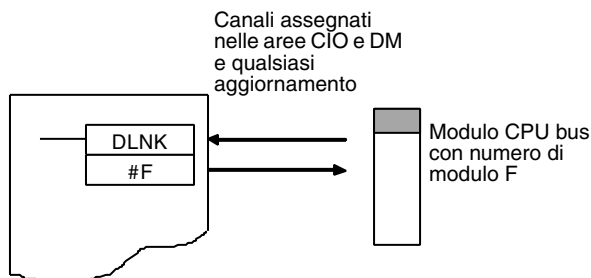
Nota Non è possibile installare i Moduli di I/O C200H nei PLC della serie CJ.

Aggiornamento dei Moduli tramite istruzione IORF(097) o DLNK(226)

L'istruzione I/O REFRESH (IORF(097)), che consente di aggiornare i dati di I/O reali in un intervallo di canali specificato, è disponibile come istruzione speciale. Grazie a tale istruzione, durante un ciclo è possibile aggiornare tutti di dati di I/O reali o solo un intervallo specificato di dati. È inoltre possibile utilizzare l'istruzione IORF per aggiornare i canali assegnati ai Moduli di I/O speciali.

L'istruzione CPU BUS UNIT REFRESH (DLNK(226)) consente invece di aggiornare i canali assegnati ai Moduli CPU bus nelle aree CIO e DM nonché di eseguire un aggiornamento speciale per il Modulo, ad esempio l'aggiornamento dei data link. L'istruzione DLNK(226) è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Aggiornamento dei Moduli tramite istruzione IORF(097)

Posizione	Sistema CPU o sistema di I/O di espansione (esclusi sistemi slave SYSMAC BUS)		
Moduli	Moduli di I/O di base	Moduli di I/O di base serie CS/CJ	Aggiornati
		Moduli di I/O di base C200H	Aggiornati
		Moduli di I/O ad alta densità C200H gruppo 2	Aggiornati
	Moduli di I/O speciale		Aggiornati
	Moduli CPU bus		Non aggiornati

Aggiornamento dei Moduli tramite istruzione DLNK(226)

Posizione	Sistema CPU o sistema di I/O di espansione (esclusi sistemi slave SYSMAC BUS)	
Moduli	Moduli di I/O di base	Non aggiornati
	Moduli di I/O speciale	Non aggiornati
	Moduli CPU bus Canali assegnati al Modulo nell'area CIO Canali assegnati al Modulo nell'area DM Aggiornamento speciale per il Modulo (data link per i Moduli Controller Link e SYSMAC LINK o I/O remoti per i Moduli DeviceNet)	Aggiornati

2-1-11 Capacità di programma

Nella tabella riportata di seguito vengono fornite le capacità massime di programma relative alle CPU delle serie CS e CJ per tutti i programmi utente, ossia la capacità totale di tutti i task. Tutte le capacità sono indicate come numero massimo di step. Se si supera la capacità, la scrittura del programma verrà disattivata.

La lunghezza di ciascuna istruzione è compresa tra 1 e 7 step. Per informazioni sul numero specifico di step in ciascuna istruzione, fare riferimento al capitolo *10-5 Tempi di esecuzione delle istruzioni e numero di step* nel *Manuale dell'operatore*. La lunghezza di ciascuna istruzione viene aumentata di uno step qualora di utilizzi un operando a doppia lunghezza.

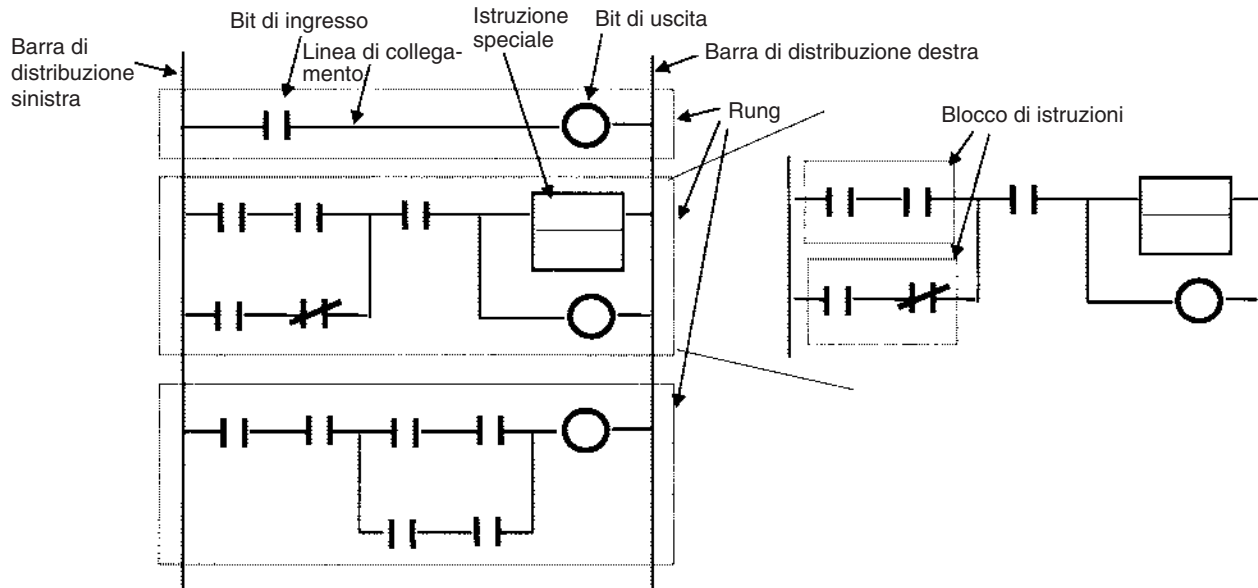
Serie	CPU	Capacità massima di programma	Punti di I/O
Serie CS	CS1H-CPU67H/CPU67-E	250.000 step	5,120
	CS1D-CPU67H	250.000 step	
	CS1D-CPU67S	250.000 step	
	CS1H-CPU66H/CPU66-E	120.000 step	
	CS1H-CPU65H/CPU65-E	60.000 step	
	CS1D-CPU65H	60.000 step	
	CS1D-CPU65S	60.000 step	
	CS1H-CPU64H/CPU64-E	30.000 step	
	CS1H-CPU63H/CPU63-E	20.000 step	
	CS1G-CPU45H/CPU45-E	60.000 step	
	CS1G-CPU44H/CPU44-E	30.000 step	1,280
	CS1D-CPU44S	30.000 step	
	CS1G-CPU43H/CPU43-E	20.000 step	960
	CS1G-CPU42H/CPU42-E	10.000 step	
	CS1D-CPU42S	10.000 step	
Serie CJ	CJ1H-CPU67H	250.000 step	2,560
	CJ1H-CPU66H	120.000 step	
	CJ1H-CPU65H	60.000 step	
	CJ1G-CPU45H/CPU45	60.000 step	1280
	CJ1G-CPU44H/CPU44	30.000 step	
	CJ1G-CPU43H	20.000 step	960
	CJ1G-CPU42H	10.000 step	
	CJ1M-CPU23/CPU13	20.000 step	640
	CJ1M-CPU22/CPU12	10.000 step	320
	CJ1M-CPU11/CPU21	5.000 step	160

Nota La capacità di memoria per i PLC delle serie CS e CJ viene misurata in step, mentre la capacità di memoria per i PLC OMRON precedenti, quali i PLC C200HX/HG/HE e della serie CV, veniva misurata in canali. Per indicazioni sulla conversione delle capacità di programma dei PLC OMRON precedenti, fare riferimento alle informazioni fornite nella parte finale del capitolo *10-5 Tempi di esecuzione delle istruzioni e numero di step* nel *Manuale dell'operatore* del PLC in uso.

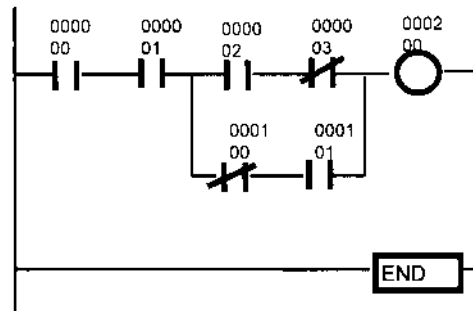
2-1-12 Concetti di base sulla programmazione ladder

Le istruzioni vengono eseguite nell'ordine elencato in memoria (ordine mnemonico). È necessario che i concetti di base sulla programmazione, così come l'ordine di esecuzione, siano corretti.

Struttura generale del diagramma ladder

Un diagramma ladder è costituito da barre di distribuzione a destra e sinistra, linee di collegamento, bit di ingresso e di uscita e istruzioni speciali. Un programma è costituito da uno o più rung di programma. Il rung di programma è un'unità che può essere ripartita quando il bus viene diviso in senso orizzontale e, dal punto di vista mnemonico, rappresenta tutte le istruzioni comprese tra un'istruzione LD/LD NOT e l'istruzione di uscita che precede immediatamente le istruzioni LD/LD NOT successive. Il rung di un programma è costituito da blocchi di istruzioni che iniziano con un'istruzione LD/LD NOT indicante un inizio logico.

Codici mnemonici

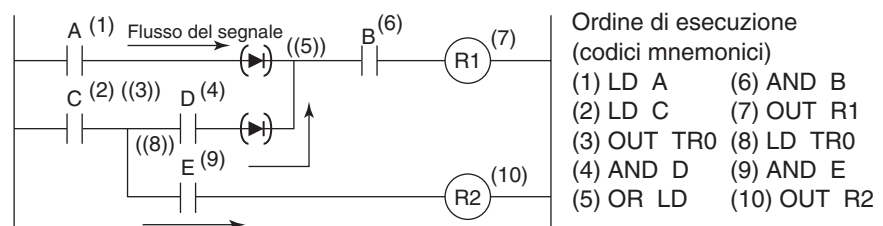
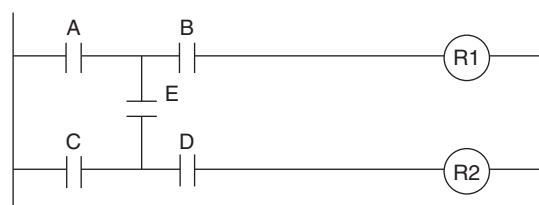
Un programma mnemonico è dato da una serie di istruzioni ladder fornite in forma mnemonica ed è costituito da indirizzi di programma. Un indirizzo di programma equivale a un'istruzione e contiene sei cifre, a partire da 000000.

Esempio

Indirizzo di programma	Istruzione (codice mnemonico)	Operando
000000	LD	000000
000001	AND	000001
000002	LD	000002
000003	AND NOT	000003
000004	LD NOT	000100
000005	AND	000101
000006	OR LD	
000007	AND LD	
000008	OUT	000200
000009	END	

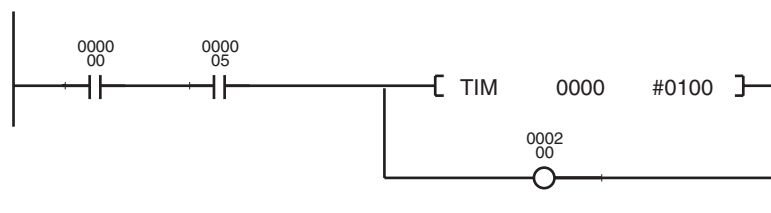
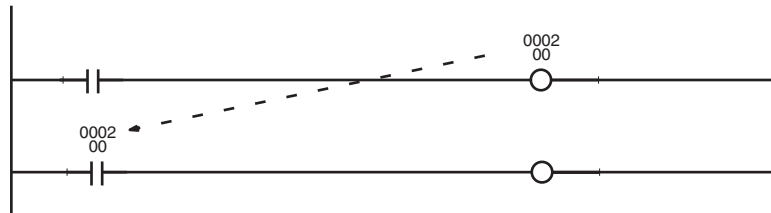
Concetti di base relativi ai programmi ladder

1,2,3...

1. Quando i diagrammi ladder vengono eseguiti dai PLC, il flusso del segnale (flusso logico di alimentazione) è sempre da sinistra a destra. Non è possibile utilizzare una programmazione che richieda un flusso logico di alimentazione da destra a sinistra. Il flusso risulta quindi diverso rispetto a quando i circuiti sono costituiti da relè di controllo cablati. Ad esempio, se il circuito "a" viene implementato in un programma PLC, l'alimentazione fluisce come se i diodi tra parentesi fossero inseriti e la bobina R2 non potesse essere azionata tramite il contatto D incluso. L'effettivo ordine di esecuzione è indicato dai codici mnemonici sulla destra. Per ottenere un funzionamento senza tali diodi immaginari, è necessario riscrivere il circuito. Inoltre, neanche il flusso logico di alimentazione "b" può essere programmato direttamente e deve quindi essere riscritto.

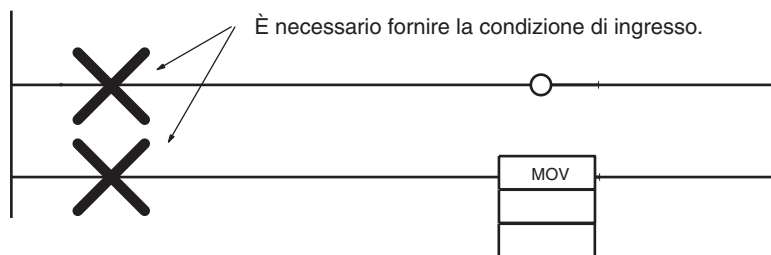
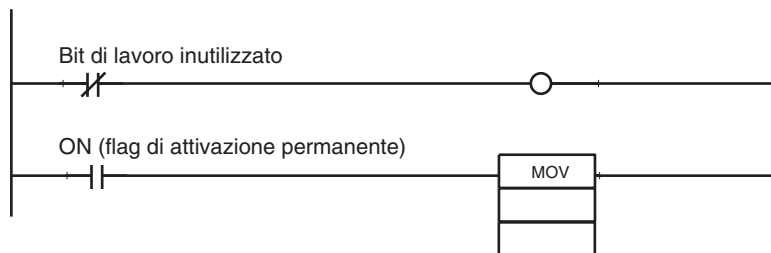
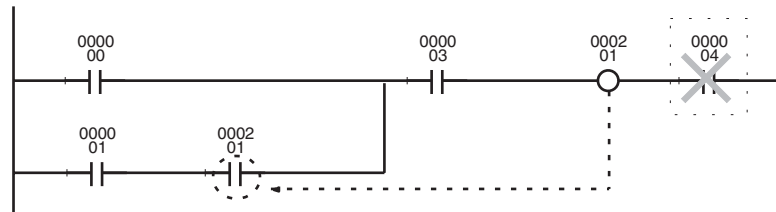
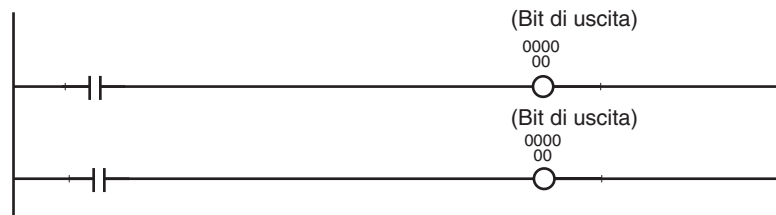
   Circuito "a"

   Circuito "b"

   Nel circuito "a" la bobina R2 non può essere azionata tramite il contatto D incluso.

   Nel circuito "b" il contatto E incluso non può essere scritto in un diagramma ladder. È quindi necessario riscrivere il programma.

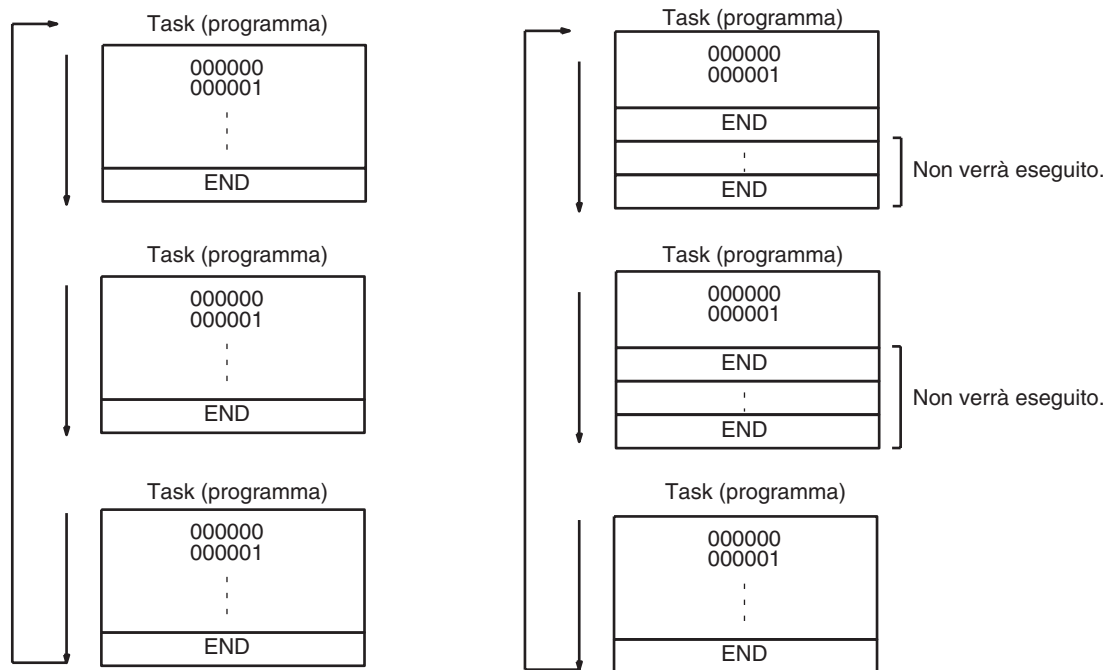
2. Non esistono limiti al numero di bit di I/O, bit di lavoro, temporizzatori e altri bit di ingresso che è possibile utilizzare. Tuttavia, è consigliabile mantenere i rung il più possibile semplici e chiari in modo da agevolarne la comprensione e la gestione, anche se ciò implica l'utilizzo di un maggior numero di bit di ingresso.
3. Non esistono limiti al numero di bit di ingresso che è possibile collegare in serie o in parallelo all'interno di rung anch'essi in serie o in parallelo.
4. È possibile collegare in parallelo due o più bit di uscita.

5. I bit di uscita possono anche essere utilizzati come bit di ingresso.

Restrizioni

1,2,3...

1. È necessario chiudere il programma ladder in modo da consentire il passaggio dei segnali (flusso logico di alimentazione) dalla barra di distribuzione sinistra alla barra di distribuzione destra. Se il programma non viene chiuso, si verificherà un errore di rung. Sarà tuttavia possibile eseguire il programma.

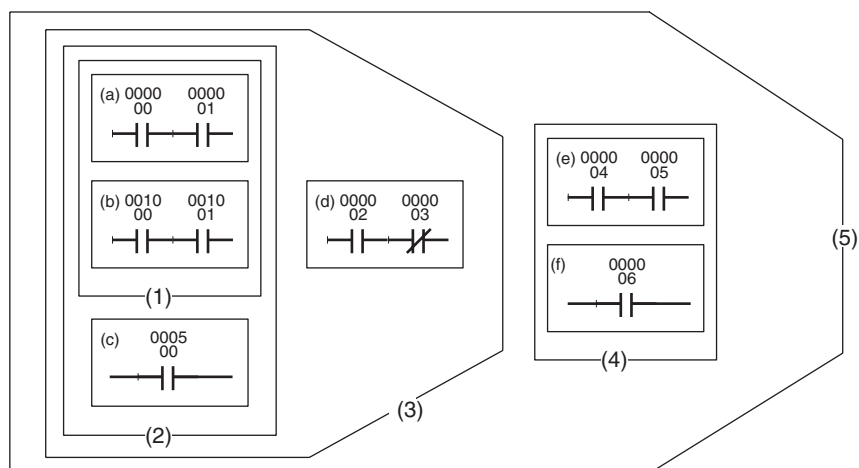
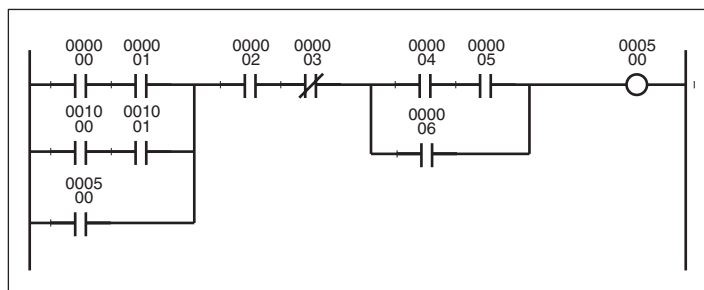
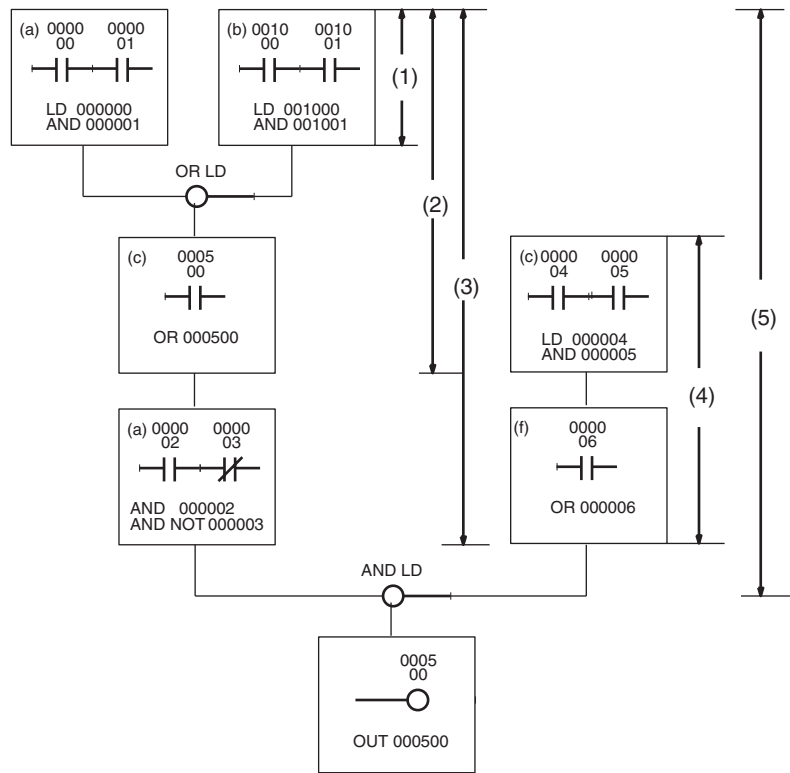
2. Non è possibile collegare bit di uscita, temporizzatori, contatori e altre istruzioni di uscita direttamente alla barra di distribuzione sinistra. Se uno di questi elementi viene collegato direttamente alla barra di distribuzione sinistra, durante la verifica del programma mediante un dispositivo di programmazione si verificherà un errore di rung. Sarà possibile eseguire il programma ma le istruzioni OUT e MOV(021) non verranno eseguite.

Se è necessario mantenere sempre attivato l'ingresso, inserire un bit di lavoro NC non utilizzato o un flag di condizione ON (flag di attivazione permanente).

3. È necessario che il bit di ingresso venga inserito sempre prima e mai dopo un'istruzione di uscita, al contrario dei bit di uscita. Se viene inserito dopo un'istruzione di uscita, durante la verifica del programma mediante un dispositivo di programmazione si verificherà un errore di posizione.

4. Non è possibile programmare lo stesso bit di uscita in un'istruzione di uscita più di una volta. Le istruzioni di un programma ladder vengono eseguite nell'ordine a partire dal primo rung in un singolo ciclo, in modo che il risultato dell'istruzione di uscita nei rung inferiori venga riflesso nel bit di uscita e che i risultati delle eventuali istruzioni precedenti che controllano il medesimo bit vengano sovrascritti e non emessi.

5. Non è possibile utilizzare un bit di ingresso in un'istruzione OUTPUT (OUT).

6. È necessario inserire un'istruzione END(001) al termine del programma in ogni task.
   - Se viene avviata l'esecuzione di un programma privo di istruzione END(001), si verificherà un errore di programma indicante l'assenza dell'istruzione END, l'indicatore ERR/ALM sul pannello anteriore della CPU si accenderà e il programma non verrà eseguito.
   - Se vengono inserite più istruzioni END(001), il programma verrà eseguito solo fino alla prima istruzione END(001).

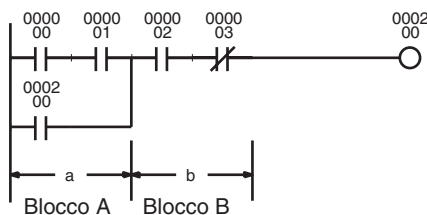
- Per rendere più agevole il debug dei programmi, è possibile inserire un'istruzione END(001) in diversi punti di interruzione nella sequenza di rung, quindi eliminare le istruzioni END(001) intermedie dopo la verifica del programma.

2-1-13 Immissione di codici mnemonici

Un inizio logico viene creato utilizzando un'istruzione LD/LD NOT. L'area compresa tra l'inizio logico e l'istruzione immediatamente precedente all'istruzione LD/LD NOT successiva viene considerata come singolo blocco di istruzioni.

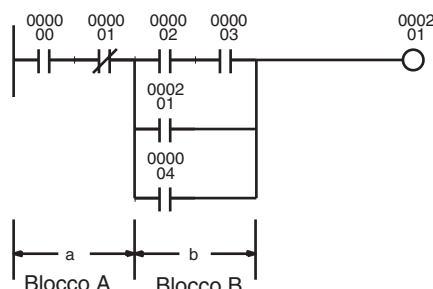
Creare un singolo rung costituito da due blocchi di istruzioni e utilizzare un'istruzione AND LD per collegare i blocchi con un AND logico o un'istruzione OR LD per collegare i blocchi con un OR logico. La procedura di immissione dei codici mnemonici (ordine e riepilogo dei rung) è illustrata nell'esempio di rung complesso fornito di seguito.

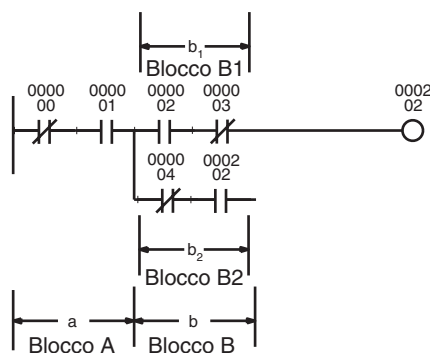
1,2,3... 1. Suddividere innanzitutto il rung in blocchi di piccole dimensioni da (a) a (f).

- Programmare i blocchi dall'alto verso il basso e da sinistra a destra.

	Indirizzo	Istruzione	Operando
(a)	000200	LD	000000
	000201	AND	000001
(b)	000202	LD	001000
	000203	AND	001001
	000204	OR LD	---
(c)	000205	OR	000500
(d)	000206	AND	000002
	000207	AND NOT	000003
(e)	000208	LD	000004
	000209	AND	000005
(f)	000210	OR	000006
	000211	AND LD	---
	000212	OUT	000500

2-1-14 Esempi di programmazione

1,2,3... 1. Rung in parallelo/in serie

Istruzione	Operandi	
LD	000000	a
AND	000001	
OR	000200	
AND	000002	b
AND NOT	000003	
OUT	000200	

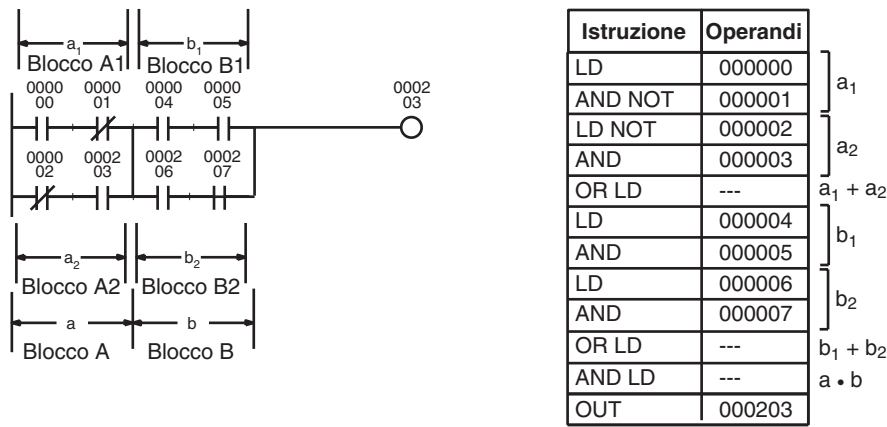
Programmare le istruzioni parallele nel blocco A e nel blocco B.

2. Rung in serie/in parallelo

Istruzione	Operandi	
LD	000000	a
AND NOT	000001	
LD	000002	b
AND	000003	
OR	000201	
OR	000004	
AND LD	---	
OUT	000201	

- Separare il rung nei blocchi A e B e programmare ciascuno blocco singolarmente.
- Collegare i blocchi A e B con un'istruzione AND LD.
- Programmare il blocco A.

Istruzione	Operandi	
LD NOT	000000	a
AND	000001	
LD	000002	b_1
AND NOT	000003	
LD NOT	000004	b_2
AND	000202	
OR LD	---	$b_1 + b_2$
AND LD	---	$a \bullet b$
OUT	000202	

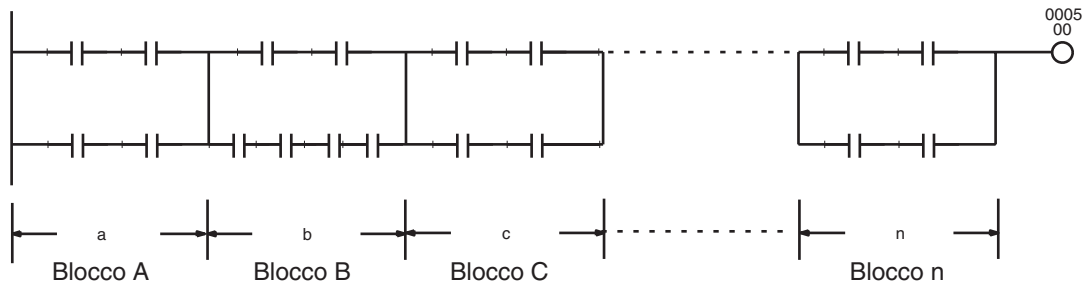
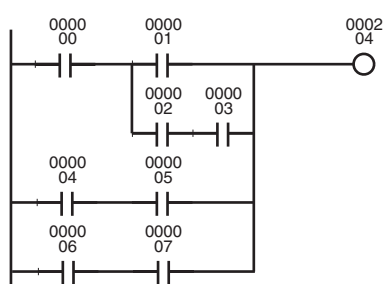
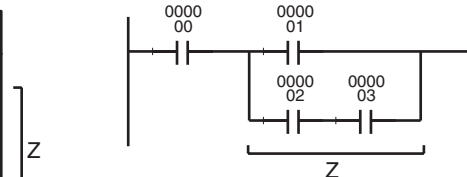
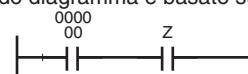
- Programmare il blocco B1 e poi il blocco B2.
- Collegare i blocchi B1 e B2 con un'istruzione OR LD, quindi collegare i blocchi A e B con un'istruzione AND LD.

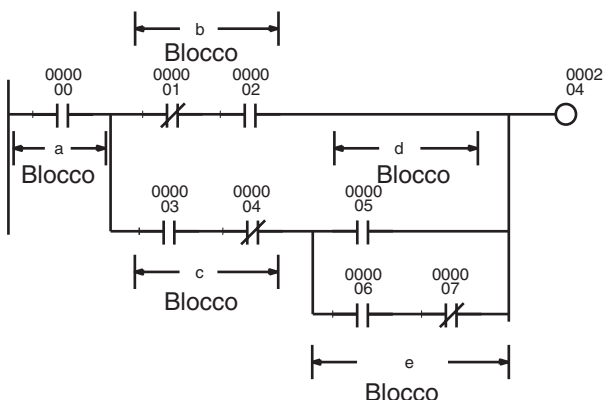
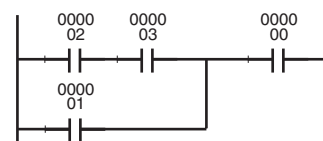
3. Esempio di collegamento in serie all'interno di un rung in serie

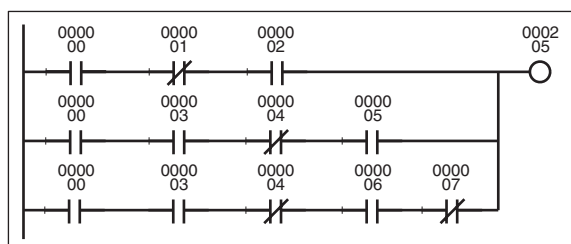
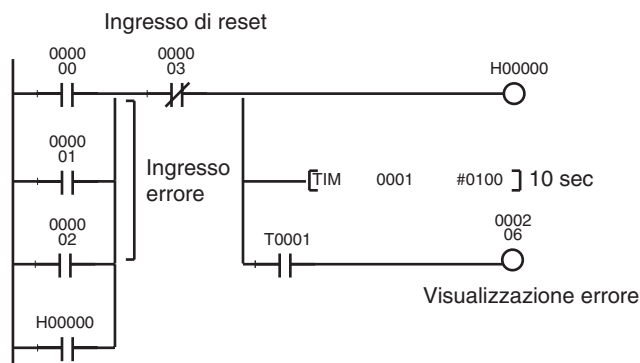
Istruzione	Operandi	
LD	000000	a_1
AND NOT	000001	
LD NOT	000002	a_2
AND	000003	
OR LD	---	$a_1 + a_2$
LD	000004	b_1
AND	000005	
LD	000006	b_2
AND	000007	
OR LD	---	$b_1 + b_2$
AND LD	---	$a \bullet b$
OUT	000203	

Programmare il blocco A_1, programmare il blocco A_2, quindi collegare i blocchi A_1 e A_2 con un'istruzione OR LD.

Programmare i blocchi B_1 e B_2 nello stesso modo.

Collegare i blocchi A e B con un'istruzione AND LD.

Ripetere per tutti i blocchi, dal blocco A al blocco n.

4. Rung complessi

Istruzione	Operando	
LD	000000	
LD	000001	Z
LD	000002	
AND	000003	
OR LD	---	
AND LD	---	
LD	000004	
AND	000005	
OR LD	---	
LD	000006	
AND	000007	
OR LD	---	
OUT	000204	

Il secondo diagramma è basato sul primo.

0000 00 Z

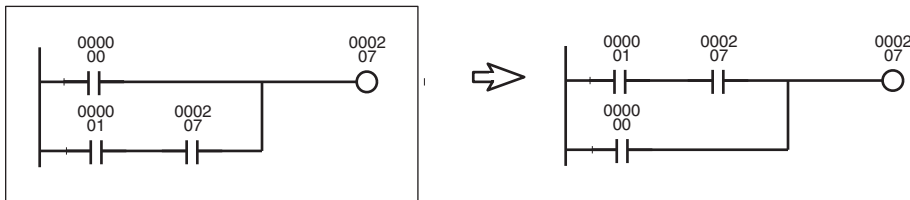
È possibile riscrivere un programma più semplice come illustrato di seguito.

È possibile riscrivere il rung precedente come illustrato di seguito.

0000 00 · 0000 01 · 0000 02 · 0002 05
0000 00 · 0000 03 · 0000 04 · 0000 05
0000 00 · 0000 03 · 0000 04 · 0000 06 · 0000 07

Istruzione	Operando	
LD	000000	a
LD NOT	000001	b
AND	000002	
LD	000003	c
AND NOT	000004	
LD	000005	d
LD	000006	
AND NOT	000007	e
OR LD	---	
AND LD	---	d + e
OR LD	---	(d + e) · c
AND LD	---	(d + e) · c + b
OUT	000205	((d + e) · c + b) · a

Istruzione	Operando
LD	000000
OR	000001
OR	000002
OR	H00000
AND NOT	000003
OUT	H00000
TIM	0001
	0100
AND	T0001
OUT	000206

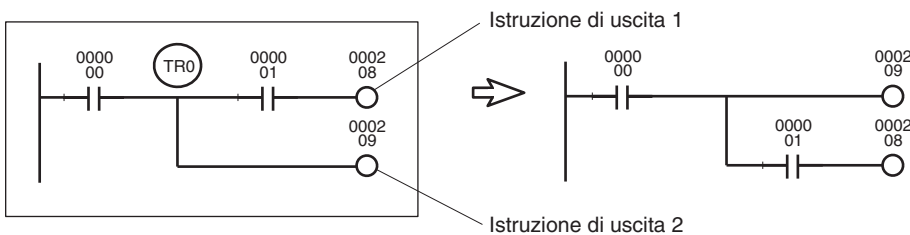
Se si utilizza un bit a ritenzione, lo stato ON/OFF verrà mantenuto in memoria anche in caso di spegnimento e alla riaccensione il segnale di errore sarà ancora presente.

5. Rung che richiedono particolare cautela o riscrittura

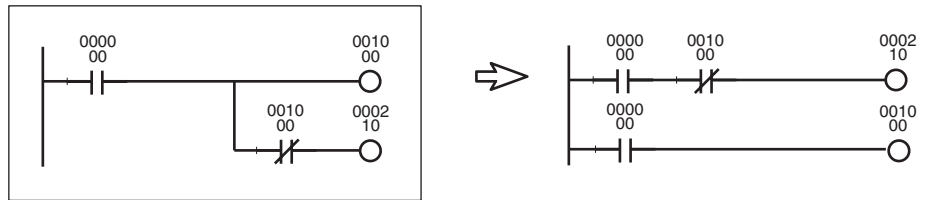
Istruzioni OR e OL LD

Con un'istruzione OR o OR NOT, viene eseguito un OR dei risultati della logica ladder inviati dall'istruzione LD o LD NOT all'istruzione OR o OR NOT, in modo che i rung possano essere sovrascritti e che l'istruzione OR LD non sia necessaria.

Esempio Se i rung vengono programmati come indicato senza apportare alcuna modifica, sarà necessario usare un'istruzione OR LD. È possibile rimuovere alcuni step riscrivendo i rung come illustrato di seguito.

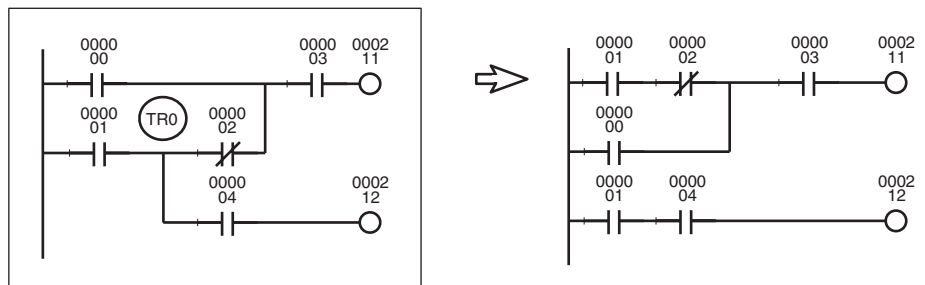
Diramazioni di istruzioni di uscita

È necessario usare un bit TR in presenza di una diramazione prima di un'istruzione AND o AND NOT. Non è però necessario ricorrere all'uso di un bit TR se il punto di diramazione è collegato direttamente alle istruzioni di uscita e l'istruzione AND o AND NOT o le istruzioni di uscita possono continuare senza modifiche.

Esempio Se i rung vengono programmati senza apportare alcuna modifica, nel punto di diramazione sarà necessario usare un'istruzione di caricamento (LD) e un'istruzione di uscita per il bit TR0 di memorizzazione temporanea. È possibile rimuovere alcuni step riscrivendo i rung.

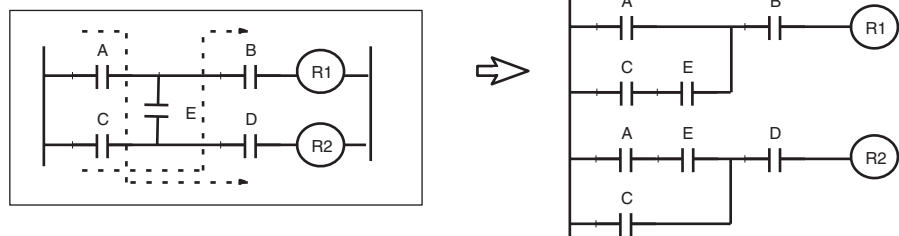
Ordine di esecuzione in base ai codici mnemonici

I PLC eseguono i programmi ladder nell'ordine di immissione dei codici mnemonici, quindi le istruzioni potrebbero avere effetti inattesi, a seconda delle modalità di scrittura dei rung. Quando si scrivono diagrammi ladder, tenere sempre presente l'ordine di esecuzione dei codici mnemonici.

Esempio Nel diagramma precedente non è possibile emettere CIO 000210, ma è possibile attivare CIO 000210 per un ciclo riscrivendo il rung, come illustrato sopra.

Rung che devono essere riscritti

I PLC eseguono le istruzioni nell'ordine di immissione dei codici mnemonici, quindi il flusso del segnale (flusso logico di alimentazione) risulta da sinistra a destra nel diagramma ladder. Non è possibile utilizzare una programmazione con flusso logico di alimentazione da destra a sinistra.

Esempio È però possibile scrivere il programma come illustrato nel diagramma a sinistra dove TR0 riceve il punto di diramazione. È tuttavia possibile ottenere il medesimo valore con i rung a destra, che risultano di più facile comprensione. È quindi consigliabile riscrivere i rung sulla sinistra come illustrato a destra.

Riscrivere i rung illustrati di seguito sulla sinistra in quanto non è possibile eseguirli.

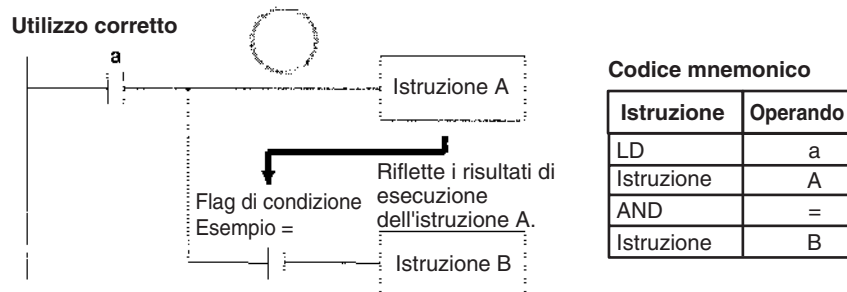
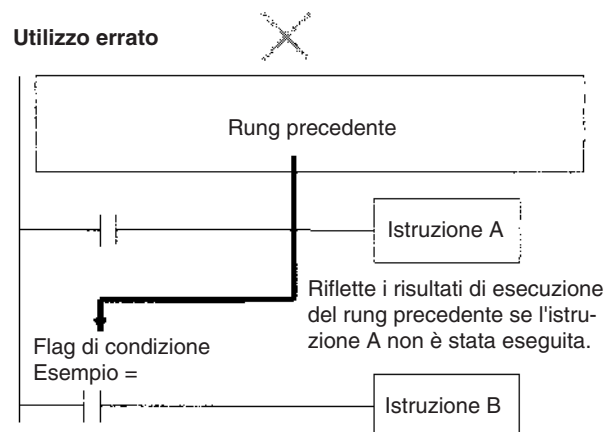
Le frecce illustrano il flusso del segnale (flusso logico di alimentazione) quando i rung sono costituiti da relè di controllo.

2-2 Precauzioni

2-2-1 Flag di condizione

Utilizzo dei flag di condizione

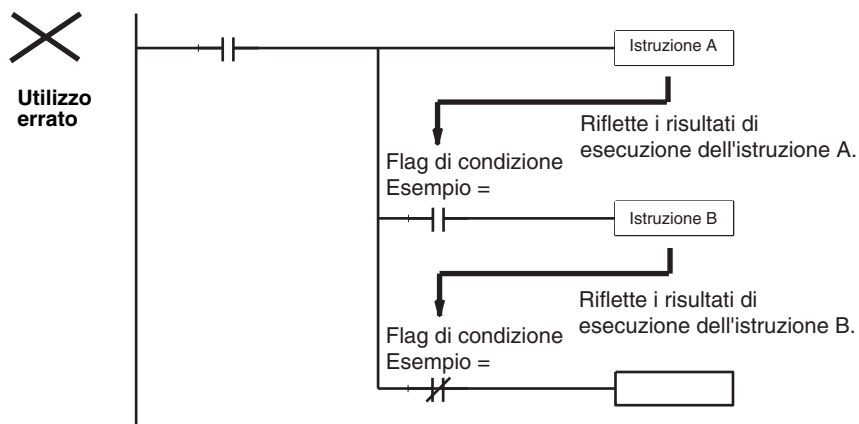
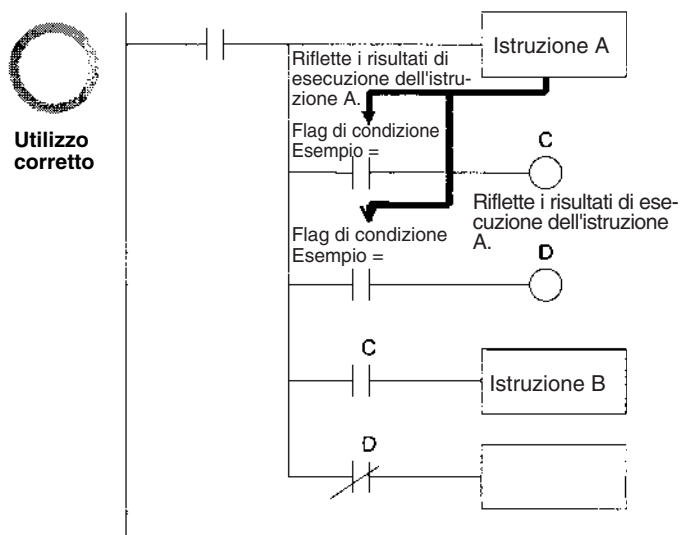
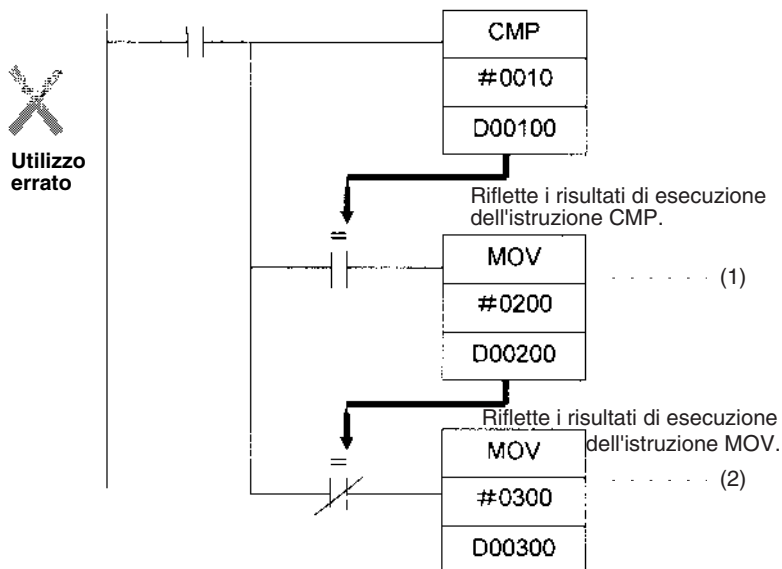
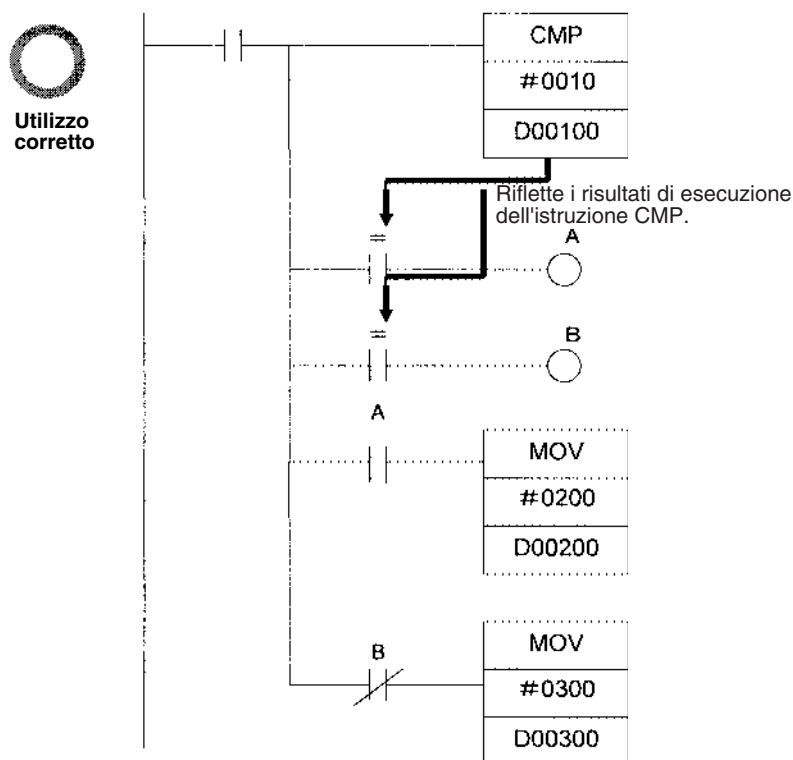
I flag di condizione sono condivisi da tutte le istruzioni e possono variare durante un ciclo a seconda dei risultati di esecuzione delle singole istruzioni. Assicurarsi pertanto di utilizzare i flag di condizione su un'uscita con diramazione avente la stessa condizione di esecuzione immediatamente dopo un'istruzione, in modo da riflettere i risultati di esecuzione di tale istruzione. Non collegare mai un flag di condizione direttamente a una barra di distribuzione, in quanto verrebbero riflessi i risultati di esecuzione di altre istruzioni.

Esempio Utilizzo dei risultati di esecuzione dell'istruzione A

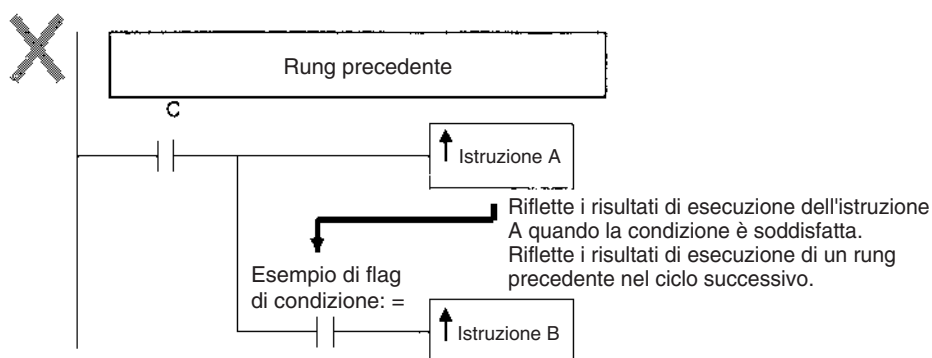
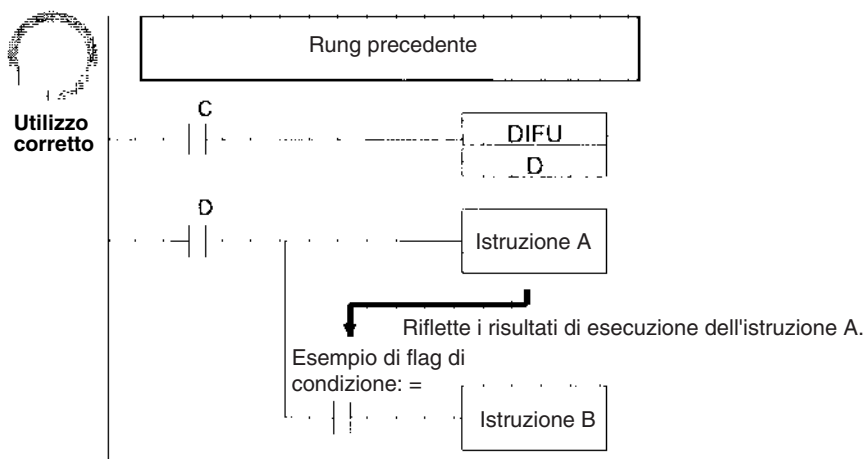
Istruzione	Operando
LD	a
Istruzione	A
AND	=
Istruzione	B

Affinché l'istruzione B venga eseguita sulla base dei risultati di esecuzione dell'istruzione A, viene utilizzata la stessa condizione di esecuzione (a) per le istruzioni A e B. In questo caso, l'istruzione B verrà eseguita in base al flag di condizione soltanto se viene eseguita l'istruzione A.

Se il flag di condizione viene collegato direttamente alla barra di distribuzione sinistra, l'istruzione B verrà eseguita sulla base dei risultati di esecuzione di un rung precedente qualora l'istruzione A non venga eseguita.

Nota I flag di condizione vengono utilizzati da tutte le istruzioni all'interno di un singolo programma (task) ma vengono cancellati quando si passa a un nuovo task. Pertanto, i risultati di esecuzione del task precedente non verranno riflessi nei task successivi. Poiché i flag di condizione sono condivisi da tutte le istruzioni, è essenziale garantire che non vi siano interferenze reciproche all'interno di un singolo programma a diagramma ladder. Di seguito viene fornito un esempio.

Utilizzo dei risultati di esecuzione negli ingressi NA e NC

Sebbene i bit di ingresso NC e NA vengano eseguiti dallo stesso ramo di uscita, i flag di condizione assumeranno i risultati di esecuzione dell'istruzione B, come illustrato nell'esempio riportato di seguito.

Accertarsi che ciascun risultato venga assunto una sola volta da un'istruzione OUTPUT per garantire che non vengano assunti i risultati di esecuzione dell'istruzione B.

Ad esempio, Nell'esempio seguente #0200 viene spostato in D00200 se D00100 contiene #0010, e #0300 viene spostato in D00300 se D00100 non contiene #0010.

Se nel rung sopra indicato D00100 contiene #0010, viene attivato il flag di uguaglianza e #0200 viene spostato in D00200 per l'istruzione al punto (1). Successivamente, però, il flag di uguaglianza viene disattivato in quanto i dati di origine #0200 non corrispondono a 0000 esadecimale. L'istruzione MOV al punto (2) verrà quindi eseguita e #0300 verrà spostato in D0300. Per impedire che vengano assunti i risultati di esecuzione relativi alla prima istruzione MOVE, è pertanto necessario inserire un rung come illustrato di seguito.

Utilizzo dei risultati di esecuzione di istruzioni differenziate

Per quanto riguarda le istruzioni differenziate, i risultati di esecuzione delle istruzioni vengono riflessi nei flag di condizione solo quando viene soddisfatta la condizione di esecuzione, mentre nei flag di condizione del ciclo successivo verranno riflessi i risultati relativi a un rung precedente anziché i risultati di esecuzione delle istruzioni differenziate. Occorre pertanto conoscere il comportamento dei flag di condizione nel ciclo successivo nel caso in cui sia necessario utilizzare i risultati di esecuzione delle istruzioni differenziate.

Nell'esempio riportato di seguito le istruzioni A e B verranno eseguite solo se viene soddisfatta la condizione di esecuzione C. Tuttavia, quando l'istruzione B assume i risultati di esecuzione dell'istruzione A, si verifica il seguente errore: se la condizione di esecuzione C rimane attiva nel ciclo successivo dopo l'esecuzione dell'istruzione A, quando lo stato del flag di condizione passa da OFF a ON a causa dei risultati riflessi da un rung precedente, l'istruzione B verrà inaspettatamente eseguita (in base alla condizione di esecuzione).

In questo caso, invece, le istruzioni A e B non sono istruzioni differenziate e viene quindi utilizzata l'istruzione DIFU (o DIFD) per determinare la differenziazione up (o down) per entrambe le istruzioni A e B, le quali verranno eseguite per un solo ciclo.

Nota Le CPU CS1-H, CJ1-H, CJ1M o CS1D supportano istruzioni per il salvataggio e il caricamento dello stato dei flag di condizione, rispettivamente CCS(282) e CCL(283). Tali istruzioni consentono di accedere allo stato dei flag di condizione da altre posizioni all'interno di un task o in un task diverso.

Principali condizioni di attivazione dei flag di condizione

Flag di errore

Il flag ER viene attivato in condizioni speciali, ad esempio quando i dati degli operandi di un'istruzione non sono corretti. Quando il flag ER è attivato, l'istruzione non viene eseguita.

Quando il flag ER è attivato, lo stato degli altri flag di condizione, quali i flag <, >, OF e UF, rimane invariato e lo stato dei flag = e N varierà in base alle istruzioni.

Per informazioni sulle condizioni che causano l'attivazione del flag ER, fare riferimento alla descrizione delle singole istruzioni nel manuale *CS/CJ-series Programmable Controllers Instructions Reference Manual (W340)*. È necessario prestare particolare attenzione in quanto alcune istruzioni comportano la disattivazione del flag ER indipendentemente dalle condizioni.

Nota Le impostazioni di configurazione del PLC relative alle condizioni di errore delle istruzioni determinano se il funzionamento debba essere interrotto quando il flag ER viene attivato. Per impostazione predefinita, il funzionamento non viene interrotto all'attivazione del flag ER. Se viene specificato di interrompere il funzionamento all'attivazione del flag ER e il funzionamento si interrompe (comportamento considerato come errore di programma), l'indirizzo del programma corrispondente al punto in cui il funzionamento è stato interrotto verrà memorizzato in A298 e A299. Contemporaneamente, verrà attivato il flag A29508 (flag di errore di elaborazione istruzione).

flag di uguaglianza

Il flag di uguaglianza è un flag temporaneo per tutte le istruzioni, tranne quando i risultati di confronto sono uguali (=). Viene impostato automaticamente dal sistema e modificato di conseguenza. È possibile disattivare (o attivare) un flag di uguaglianza mediante un'istruzione dopo che è stato attivato (o disattivato) da un'istruzione precedente. Il flag di uguaglianza viene ad esempio attivato quando 0000 esadecimale viene spostato come dato di origine mediante l'istruzione MOV o un'altra istruzione di spostamento, e viene disattivato in tutte le altre condizioni. Anche se il flag di uguaglianza viene attivato da un'istruzione, l'istruzione di spostamento verrà eseguita immediatamente e il flag di uguaglianza verrà attivato o disattivato a seconda che i dati di origine per l'istruzione di spostamento corrispondano o meno a 0000 esadecimale.

Flag di riporto

Il flag CY viene utilizzato nelle istruzioni di scorrimento, nelle istruzioni di addizione e sottrazione con ingresso di riporto, nei prestiti e nei riporti delle istruzioni di addizione e sottrazione, nonché nelle istruzioni del Modulo di I/O speciale, nelle istruzioni PID e nelle istruzioni FPD. Osservare le precauzioni riportate di seguito.

Nota

1. È possibile che il flag CY rimanga attivato (o disattivato) in seguito ai risultati di esecuzione di una determinata istruzione e venga successivamente utilizzato in altre istruzioni (un'istruzione di addizione e sottrazione con riporto o un'istruzione di scorrimento). Assicurarsi di cancellare il flag di riporto quando necessario.
2. Il flag CY può venire attivato (o disattivato) in base ai risultati di esecuzione di una determinata istruzione e venire disattivato (o attivato) da un'altra istruzione. Accertarsi che il flag di riporto rifletta i risultati corretti quando lo si utilizza.

Flag < (minore di) e flag > (maggiore di)

I flag < e > vengono utilizzati nelle istruzioni di confronto, oltre che in LMT, BAND, ZONE, PID e in altre istruzioni.
Il flag < o > può essere disattivato (o attivato) da un'altra istruzione anche se è stato attivato (o disattivato) in base ai risultati dell'esecuzione di una determinata istruzione.

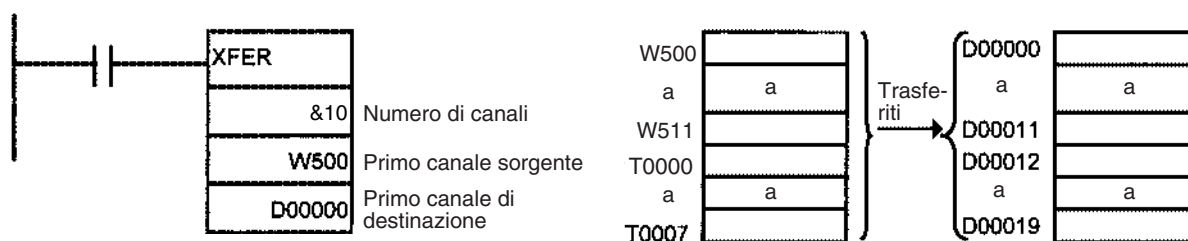
Flag negativo

Il flag N viene disattivato quando il bit all'estrema sinistra del canale dei risultati di esecuzione di un'istruzione è "1" per determinate istruzioni e viene disattivato incondizionatamente per altre istruzioni.

Specifica degli operandi per più canali

Con i PLC delle serie CS e CJ, un'istruzione viene eseguita così come è stata scritta, anche se viene specificato un operando che richiede più canali e i canali si trovano in aree diverse. In questo caso, i canali verranno presi in base all'ordine degli indirizzi di memoria del PLC. Il flag di errore **non** verrà attivato.

Si considerino ad esempio i risultati dell'esecuzione di un trasferimento di blocchi mediante l'istruzione XFER(070), per il quale vengono specificati 20 canali a partire da W500. In tal caso, il limite dell'area di lavoro, che termina con il canale W511, viene superato, ma l'istruzione verrà eseguita senza attivare il flag di errore. Negli indirizzi di memoria del PLC i valori attuali dei temporizzatori vengono mantenuti in memoria nei canali successivi all'area di lavoro e, di conseguenza, in base all'istruzione, i valori nei canali da W500 a W511 verranno trasferiti negli indirizzi da D00000 a D00011 mentre i valori attuali dell'intervallo da T0000 a T0007 verranno trasferiti negli indirizzi da D00012 a D00019.

Nota Per informazioni su indirizzi di memoria specifici del PLC, fare riferimento all'appendice *Mappa degli indirizzi di memoria del PLC*.

2-2-2 Sezioni di programma speciali

I programmi per i PLC delle serie CS e CJ contengono sezioni speciali che consentono di controllare le condizioni delle istruzioni. Di seguito sono riportate le sezioni di programma speciali disponibili.

Sezione di programma	Istruzioni	Condizione di istruzione	Stato
Subroutine	Istruzioni SBS, SBN e RET	Viene eseguito il programma di subroutine.	Viene eseguita la sezione di programma di subroutine compresa tra le istruzioni SBN e RET.
Sezione IL-ILC	Istruzioni IL e ILC	La sezione viene interbloccata.	I bit di uscita vengono disattivati e i temporizzatori reimpostati. Non verranno eseguite altre istruzioni e verrà mantenuto lo stato precedente.
Sezione ladder con struttura a step	Istruzioni STEP S e STEP		
Ciclo FOR-NEXT	Istruzioni FOR e NEXT	Interruzione in corso.	Esecuzione di cicli
Sezione JMP0-JME0	Istruzioni JMP0 e JME0		Salto
Sezione di programma a blocchi	Istruzioni BPRG e BEND	Viene eseguito il programma a blocchi.	Viene eseguito il programma a blocchi elencato nei codici mnemonici posizionati tra le istruzioni BPRG e BEND.

Combinazioni di istruzioni

Nella tabella riportata di seguito vengono illustrati i tipi di istruzioni speciali che è possibile utilizzare all'interno di altre sezioni del programma.

	Subroutine	Sezione IL-ILC	Sezione ladder con struttura a step	Ciclo FOR-NEXT	Sezione JMP0-JME0	Sezione di programma a blocchi
Subroutine	Non consentito	Non consentito	Non consentito	Non consentito	Non consentito	Non consentito
IL-ILC	**OK**	Non consentito	Non consentito	**OK**	**OK**	Non consentito
Sezione ladder con struttura a step	Non consentito	**OK**	Non consentito	Non consentito	**OK**	Non consentito
Ciclo FOR-NEXT	**OK**	**OK**	Non consentito	**OK**	**OK**	Non consentito
JMP0-JME0	**OK**	**OK**	Non consentito	Non consentito	Non consentito	Non consentito
Sezione di programma a blocchi	**OK**	**OK**	**OK**	Non consentito	**OK**	Non consentito

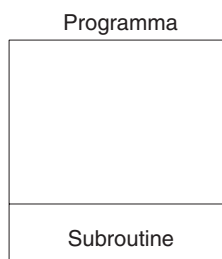
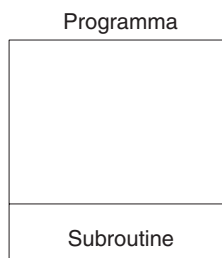
Nota Non è possibile utilizzare istruzioni che specificano aree di programma per programmi all'interno di altri task. Per ulteriori informazioni, fare riferimento al capitolo *4-2-2 Limitazioni relative alle istruzioni dei task*.

Subroutine

Tutte le subroutine devono essere raggruppate e posizionate immediatamente prima dell'istruzione END(001), ma dopo una sezione di programmazione diversa da una subroutine, in tutti i programmi. Pertanto, non è possibile inserire una subroutine all'interno di una sezione ladder con struttura a step, un programma a blocchi, un ciclo FOR–NEXT o una sezione JMP0-JME0. Se un programma diverso da una subroutine viene inserito dopo il programma di subroutine (da SBN a RET), tale programma non verrà eseguito.

Programma

Subroutine

Programma

Subroutine

Istruzioni non disponibili nelle subroutine

Non è possibile inserire in una subroutine le istruzioni riportate di seguito.

Funzione	Codice mnemonico	Istruzione
Controllo degli step di processo	STEP(008)	Definizione di una sezione ladder con struttura a step
	SNXT(009)	Esecuzione del programma ladder con struttura a step un'istruzione alla volta

Nota **Sezioni di programma a blocchi**
Una subroutine può includere una sezione di programma a blocchi. Se tuttavia il programma a blocchi è in stato di attesa (WAIT) quando si passa dall'esecuzione della subroutine all'esecuzione del programma principale, la sezione di programma a blocchi resterà in stato di attesa al successivo richiamo.

Istruzioni non disponibili nelle sezioni di programma ladder con struttura a step

Funzione	Codice mnemonico	Istruzione
Controllo della sequenza	FOR(512), NEXT(513) e BREAK(514)	FOR, NEXT e BREAK LOOP
	END (001)	END
	IL(002) e ILC(003)	INTERLOCK e INTERLOCK CLEAR
	JMP(004) e JME(005)	JUMP e JUMP END
	CJP(510) e CJPN(511)	CONDITIONAL JUMP e CONDITIONAL JUMP NOT
	JMP0(515) e JME0(516)	MULTIPLE JUMP e MULTIPLE JUMP END
Subroutine	SBN(092) e RET(093)	SUBROUTINE ENTRY e SUBROUTINE RETURN
Programmi a blocchi	IF(802) (NOT), ELSE(803) e IEND(804)	Istruzioni di diramazione
	BPRG(096) e BEND(801)	BLOCK PROGRAM BEGIN/ END
	EXIT(806) (NOT)	CONDITIONAL BLOCK EXIT (NOT)
	LOOP(809) e LEND(810) (NOT)	Controllo dei cicli
	WAIT(805) (NOT)	ONE CYCLE WAIT (NOT)
	TIMW(813)	TIMER WAIT
	TMHW(815)	HIGH-SPEED TIMER WAIT
	CNTW(814)	COUNTER WAIT
	BPPS(811) e BPRS(812)	BLOCK PROGRAM PAUSE e RESTART

Nota

1. È possibile utilizzare una sezione di programma ladder con struttura a step in una sezione di interblocco (tra IL e ILC). Quando l'interblocco viene attivato, la sezione ladder con struttura a step viene completamente reimpostata.
2. È possibile inserire una sezione di programma ladder con struttura a step tra le istruzioni MULTIPLE JUMP (JMP0) e MULTIPLE JUMP END (JME0).

Istruzioni non disponibili nelle sezioni di programma a blocchi

Non è possibile inserire nelle sezioni di programma a blocchi le istruzioni riportate di seguito.

Classificazione in base alla funzione	Codice mnemonico	Istruzione
Controllo della sequenza	FOR(512), NEXT(513) e BREAK(514)	FOR, NEXT e BREAK LOOP
	END (001)	END
	IL(002) e ILC(003)	INTERLOCK e INTERLOCK CLEAR
	JMP0(515) e JME0(516)	MULTIPLE JUMP e MULTIPLE JUMP END
Ingresso sequenza	UP(521)	CONDITION ON
	DOWN(522)	CONDITION OFF
Uscita sequenza	DIFU	DIFFERENTIATE UP
	DIFD	DIFFERENTIATE DOWN
	KEEP	KEEP
	OUT	OUTPUT
	OUT NOT	OUTPUT NOT
Temporizzatore/contatore	TIM	TIMER
	TIMH	HIGH-SPEED TIMER
	TMHH(540)	ONE-MS TIMER
	TTIM(087)	ACCUMULATIVE TIMER
	TIML(542)	LONG TIMER
	MTIM(543)	MULTI-OUTPUT TIMER
	CNT	COUNTER
	CNTR	REVERSIBLE COUNTER
Subroutine	SBN(092) e RET(093)	SUBROUTINE ENTRY e SUBROUTINE RETURN
Scorrimento di dati	SFT	SHIFT
Controllo degli step del programma ladder	STEP(008) e SNXT(009)	STEP DEFINE e STEP START
Controllo di dati	PID	PID CONTROL
Programma a blocchi	BPRG(096)	BLOCK PROGRAM BEGIN
Diagnosi dei danni	FPD(269)	FAILURE POINT DETECTION

Nota

1. È possibile utilizzare programmi a blocchi in una sezione di programma ladder con struttura a step.
2. È possibile utilizzare un programma a blocchi in una sezione di interblocco (tra IL e ILC). Quando l'interblocco viene attivato, la sezione del programma a blocchi non viene eseguita.
3. È possibile inserire una sezione di programma a blocchi tra le istruzioni MULTIPLE JUMP (JMP0) e MULTIPLE JUMP END (JME0).
4. È possibile utilizzare le istruzioni JUMP (JMP) e CONDITIONAL JUMP (CJP/CJPN) in una sezione di programma a blocchi. Le istruzioni JUMP (JMP) e JUMP END (JME), nonché le istruzioni CONDITIONAL JUMP (CJP/CJPN) e JUMP END (JME), possono essere utilizzate nella sezione di programma a blocchi esclusivamente se specificate in coppia. In caso contrario, il programma non verrà eseguito in modo corretto.

2-3 Verifica dei programmi

È possibile verificare la correttezza dei programmi per i PLC delle serie CS e CJ nelle seguenti fasi:

- Verifica degli ingressi durante le operazioni di immissione dalla Console di programmazione
- Verifica del programma tramite CX-Programmer
- Verifica delle istruzioni durante l'esecuzione
- Verifica degli errori fatali (errori di programma) durante l'esecuzione

2-3-1 Errori durante le operazioni di immissione da un dispositivo di programmazione

Console di programmazione

Durante l'immissione, sulla Console di programmazione vengono visualizzati gli errori riportati di seguito.

Visualizzazione errore	Causa
CHK MEM	Il pin 1 del DIP switch della CPU è posizionato su ON (protezione da scrittura).
IO No. ERR	Si è tentato di immettere dati di I/O non validi.

CX-Programmer

Il programma viene automaticamente verificato da CX-Programmer durante le seguenti fasi.

Operazione	Contenuto controllato
Immissione di diagrammi ladder	Ingressi di istruzioni, ingressi di operandi, sequenze di programmazione
Caricamento dei file	Tutti gli operandi per tutte le istruzioni e tutte le sequenze di programmazione
Scaricamento dei file	Modelli supportati dalla serie CS/CJ e tutti gli operandi per tutte le istruzioni
Modifica in linea	Capacità ecc.

I risultati della verifica vengono inviati alla scheda Text (Testo) della finestra di output. Inoltre, la barra di distribuzione sinistra corrispondente alle sezioni di programma non valide verrà visualizzata in rosso nella visualizzazione del diagramma ladder.

2-3-2 Verifica del programma tramite CX-Programmer

Nella tabella riportata di seguito sono elencati di errori rilevati dalla funzione di verifica dei programmi di CX-Programmer.

CX-Programmer non controlla gli errori di intervallo relativi agli operandi indirizzati in modo indiretto nelle istruzioni. Gli errori di indirizzamento indiretto verranno rilevati durante la verifica del programma in fase di esecuzione e verrà attivato il flag ER, come descritto nella sezione successiva. Per ulteriori informazioni, fare riferimento al manuale *CS/CJ-series Programmable Controllers Instructions Reference Manual (W340)*.

Quando il programma viene verificato in CX-Programmer, l'operatore può impostare i livelli di verifica A, B e C (in ordine di gravità dell'errore) nonché un livello di verifica personalizzato.

Area	Verifica
Dati non validi: diagramma ladder	Posizione delle istruzioni
	Linee di I/O
	Collegamenti
	Completezza delle istruzioni e delle operazioni
Istruzioni supportate dal PLC	Istruzioni e operandi supportati dal PLC
	Variazioni di istruzioni (NOT, !, @ e %)
	Integrità del codice oggetto
Intervalli degli operandi	Intervalli delle aree degli operandi
	Tipi di dati degli operandi
	Verifica dell'accesso ai canali di sola lettura
	Le verifiche eseguite sugli intervalli degli operandi includono: • Costanti (#, &, +, –) • Codici di controllo • Verifica dei limiti di area per gli operandi a più canali • Verifica delle dimensioni per gli operandi a più canali • Sovrapposizione degli intervalli degli operandi • Assegnazioni multicanale • Operandi a doppia lunghezza • Verifica dei limiti di area per gli offset
Capacità di programma per il PLC	Numero di step
	Capacità totale
	Numero di task
Sintassi	Verifica delle chiamate per le istruzioni in coppia • IL–ILC • JMP–JME, CJP/CJPN-JME • SBS–SBN–RET, MCRO–SBN–RET • STEP–SNXT • BPRG–BEND • IF–IEND • LOOP–LEND
	Posizioni di programmazione non consentite per BPRG–BEND
	Posizioni di programmazione non consentite per SBN–RET
	Posizioni di programmazione non consentite per STEP–SNXT
	Posizioni di programmazione non consentite per FOR–NEXT
	Posizioni di programmazione non consentite per i task ad interrupt
	Posizioni di programmazione obbligatorie per BPRG–BEND
	Posizioni di programmazione obbligatorie per FOR–NEXT
	Annidamento non valido
	Istruzione END(001)
	Coerenza dei numeri
Struttura del diagramma ladder	Overflow dello stack
Duplicazione di uscite	Verifica delle uscite duplicate • Per bit • Per canale • Istruzioni del temporizzatore/contatore • Canali lunghi (2 canali e 4 canali) • Canali multipli assegnati • Intervalli di inizio/fine • Numeri di FAL • Istruzioni con più operandi di uscita
Task	Verifica dei task impostati per l'avvio all'inizio del funzionamento
	Assegnazione di programmi di task

Nota La duplicazione delle uscite non viene verificata tra più task ma solo all'interno di singoli task.

Operandi a più canali

Durante la verifica del programma, vengono controllati anche i limiti delle aree di memoria per gli operandi a più canali, come illustrato nella tabella riportata di seguito.

CX-Programmer	Console di programmazione
Per gli operandi a più canali che superano il limite di un'area di memoria, CX-Programmer fornisce le seguenti opzioni: • Il programma non può essere trasferito sulla CPU. • Il programma non può essere letto dalla CPU. • Per la verifica del programma vengono generati errori di compilazione. • Durante la programmazione in modalità non in linea vengono visualizzati dei messaggi di avviso. • Durante la modifica in linea in modalità PROGRAM o MONITOR vengono visualizzati dei messaggi di avviso.	La verifica viene eseguita durante l'immissione dei programmi, ovvero non è possibile scrivere operandi che superano il limite di un'area di memoria.

2-3-3 Verifica del programma in fase di esecuzione

I controlli relativi alla posizione degli operandi e delle istruzioni vengono eseguiti durante le operazioni di immissione dal dispositivo di programmazione (Console di programmazione inclusa), nonché durante la verifica del programma dal dispositivo di programmazione (Console di programmazione esclusa). Tuttavia, questi controlli non sono definitivi.

Durante l'esecuzione delle istruzioni vengono effettuati i seguenti controlli.

Tipo di errore	Flag attivato in seguito all'errore	Continuazione/interruzione del funzionamento
1. Errore di elaborazione dell'istruzione	Flag ER Il flag di errore di elaborazione istruzione(A29508) verrà attivato anche se si è specificato di interrompere il funzionamento in seguito a un errore.	È possibile specificare se proseguire o interrompere il funzionamento in seguito a errori di elaborazione delle istruzioni nella configurazione del PLC. Per impostazione predefinita, il funzionamento non viene interrotto. Verrà generato un errore di programma e il funzionamento si interromperà solo se esplicitamente specificato.
2. Errore di accesso	Flag AER Il flag di errore di accesso (A29510) verrà attivato se si è specificato di interrompere il funzionamento quando si verifica un errore.	È possibile specificare se proseguire o interrompere il funzionamento in seguito a errori di elaborazione delle istruzioni nella configurazione del PLC. Per impostazione predefinita, il funzionamento non viene interrotto. Verrà generato un errore di programma e il funzionamento si interromperà solo se esplicitamente specificato.
3. Errore di istruzione illegale	Flag di errore di istruzione illegale (A29514)	Errore fatale (errore di programma)
4. Errore di overflow UM (memoria utente)	Flag di errore di overflow UM (A29515)	Errore fatale (errore di programma)

Errori di elaborazione delle istruzioni

Se durante l'esecuzione di un'istruzione vengono forniti dati non corretti o si tenta di eseguire un'istruzione all'esterno di un task, verrà generato un errore di elaborazione dell'istruzione. In tali circostanze, i dati richiesti all'inizio dell'elaborazione dell'istruzione vengono controllati e, se incorretti, l'istruzione non viene eseguita, viene attivato il flag ER (flag di errore) e i flag EQ e N possono essere mantenuti o venire disattivati a seconda dell'istruzione.

Il flag ER (flag di errore) verrà disattivato se l'istruzione, escluse le istruzioni di ingresso, viene completata correttamente. Le condizioni che comportano l'attivazione del flag ER variano a seconda delle istruzioni. Per ulteriori informazioni, fare riferimento alle descrizioni delle singole istruzioni nel manuale *CS/CJ-series Programmable Controllers Programming Manual* (W340).

Se il PLC è impostato per interrompere il funzionamento in caso di errore di istruzione, il funzionamento verrà interrotto (errore fatale) e il flag di errore di elaborazione istruzione (A29508) verrà attivato nel caso in cui si verifichi un errore di elaborazione dell'istruzione e venga attivato il flag ER.

Errori di accesso illegale

Gli errori di accesso illegale indicano che, durante l'accesso all'indirizzo in cui è specificato l'operando dell'istruzione, è stato eseguito l'accesso a un'area errata utilizzando uno dei seguenti metodi:

a) È stata eseguita un'operazione di lettura o scrittura per un'area dei parametri.
b) È stata eseguita un'operazione di scrittura in un'area di memoria non installata (vedere nota).
c) È stata eseguita un'operazione di scrittura in un'area EM specificata come memoria file nell'area EM.
d) È stata eseguita un'operazione di scrittura in un'area di sola lettura.
e) Il valore specificato in un indirizzo DM/EM indiretto in modalità BCD non era in formato BCD (ad esempio, *D000001 contiene #A000).

Se si verifica un errore di accesso, l'elaborazione delle istruzioni prosegue e non verrà attivato il flag di errore (flag ER) bensì il flag di errore di accesso (flag AER).

Nota Un errore di accesso si verifica nei seguenti casi:

- Quando un indirizzo EM specificato è superiore a 32767 (esempio: E32768) per il banco corrente.
- Il banco finale (esempio: C) viene specificato per un indirizzo EM indiretto in modalità binaria e il canale specificato contiene un valore esadecimale compreso nell'intervallo da 8000 a FFFF (esempio: @EC_00001 contiene #8000).
- Il banco corrente (esempio: C) viene specificato per un indirizzo EM indiretto in modalità binaria e il canale specificato contiene un valore esadecimale compreso nell'intervallo da 8000 a FFFF (esempio: @EC_00001 contiene #8000).
- Un registro IR contenente l'indirizzo di memoria interna di un bit viene utilizzato come indirizzo di canale o un registro IR contenente l'indirizzo di memoria interna di un canale viene utilizzato come indirizzo di bit.

Se il PLC è impostato per interrompere il funzionamento in caso di errore di istruzione, il funzionamento verrà interrotto (errore fatale) e il flag di errore di accesso illegale (A29510) verrà attivato nel caso in cui si verifichi un errore di accesso illegale e venga attivato il flag AER.

Nota Il flag di errore di accesso (flag AER) non verrà cancellato dopo l'esecuzione di un task. Se il PLC è impostato per non interrompere il funzionamento in caso di errore di istruzione, sarà possibile monitorare lo stato di questo flag fino al punto immediatamente precedente all'istruzione END(001) per determinare se si è verificato un errore di accesso illegale nel programma del task. Se il flag AER viene monitorato su una Console di programmazione, sarà possibile verificare lo stato del flag AER finale dopo l'esecuzione dell'intero programma utente.

Altri errori

Errori di istruzione illegale

Gli errori di istruzione illegale indicano che si è tentato di eseguire dati di istruzioni diversi da quelli definiti nel sistema. Di norma, se il programma viene creato con un dispositivo di programmazione della serie CS/CJ (incluse le Console di programmazione), questo errore non si verifica.

Nei rari casi in cui dovesse verificarsi, l'errore verrà considerato come errore di programma, il funzionamento verrà interrotto (errore fatale) e verrà attivato il flag di istruzione illegale (A29514).

Errori di overflow UM (memoria utente)

Gli errori di overflow UM indicano che si è tentato di eseguire dati di istruzioni memorizzati in indirizzi successivi all'ultimo indirizzo della memoria utente (UM, User Memory) definita come area di archiviazione del programma. Di norma, se il programma viene creato con un dispositivo di programmazione della serie CS/CJ (incluse le Console di programmazione), questo errore non si verifica.

Nei rari casi in cui dovesse verificarsi, l'errore verrà considerato come errore di programma, il funzionamento verrà interrotto (errore fatale) e verrà attivato il flag di overflow UM (A29515).

2-3-4 Verifica degli errori fatali

Gli errori riportati di seguito sono errori di programma fatali che comportano l'interruzione del funzionamento della CPU. Quando il funzionamento viene interrotto a causa di un errore di programma, il numero del task in cui si è verificata l'interruzione verrà memorizzato in A294 e l'indirizzo del programma verrà memorizzato in A298 e A299. È possibile determinare la causa dell'errore di programma in base alle informazioni riportate di seguito.

Indirizzo	Descrizione	Dati memorizzati
A294	Se il funzionamento viene interrotto a causa di un errore di programma, il tipo di task e il numero del task elaborato al momento in cui si è verificata l'interruzione vengono memorizzati in questa posizione. Nel caso in cui in un ciclo non siano presenti task ciclici attivi, ossia se non vi sono task ciclici da eseguire, verrà memorizzato il valore FFFF esadecimale.	Task ciclico: da 0000 a 001F esadecimale (task ciclici da 0 a 31) Tash ad interrupt: da 8000 a 80FF esadecimale (task ad interrupt da 0 a 255)
A298 e A299	Se il funzionamento viene interrotto a causa di un errore di programma, l'indirizzo del programma corrispondente al punto in cui si è verificata l'interruzione del funzionamento verrà memorizzato in questa posizione. Se manca l'istruzione END(001), e quindi viene attivato il flag A29511, verrà memorizzato l'indirizzo in cui era prevista l'istruzione END(001). Se si verifica un errore di esecuzione del task, e quindi viene attivato il flag A29512, in A298 e A299 verrà memorizzato il valore FFFFFFFF esadecimale.	A298: porzione destra dell'indirizzo del programma A299: porzione sinistra dell'indirizzo del programma

Nota Se viene attivato flag di errore o il flag di errore di accesso, l'errore verrà considerato come errore di programma e potrà essere utilizzato per interrompere l'elaborazione a livello di CPU. Specificare nella configurazione del PLC l'azione da intraprendere a fronte di errori di programma.

Errore di programma	Descrizione	Flag correlati
Istruzione END mancante	Nel programma non è presente alcuna istruzione END.	Viene attivato il flag di errore di END mancante (A29511).
Errore durante l'esecuzione dei task	Nel ciclo non vi è alcun task pronto. Non è assegnato alcun programma a un task. Il numero del task ad interrupt corrispondente non è presente anche se la condizione di esecuzione per il task ad interrupt è stata soddisfatta.	Viene attivato il flag di errore di task (29512).
Errore di elaborazione delle istruzioni (flag ER attivato) e interruzione del funzionamento in seguito a errori di istruzione impostata nella configurazione del PLC	Durante il tentativo di esecuzione di un'istruzione, nell'operando sono stati immessi valori di dati errati.	Se nella configurazione del PLC è impostata l'interruzione del funzionamento in seguito a errori di istruzione, vengono attivati il flag ER e il flag di errore di elaborazione istruzione (A29508).
Errore di accesso illegale (flag AER attivato) e interruzione del funzionamento in seguito a errori di istruzione impostata nella configurazione del PLC	È stata eseguita un'operazione di lettura o scrittura per un'area dei parametri. È stata eseguita un'operazione di scrittura in un'area di memoria non installata (vedere nota). È stata eseguita un'operazione di scrittura in un'area EM specificata come memoria file nell'area EM. È stata eseguita un'operazione di scrittura in un'area di sola lettura. Il valore specificato in un indirizzo DM/EM indiretto in modalità BCD non era in formato BCD.	Se nella configurazione del PLC è impostata l'interruzione del funzionamento in seguito a errori di istruzione, vengono attivati il flag AER e il flag di errore di accesso illegale (A29510).
Errore BCD DM/EM indiretto e interruzione del funzionamento in seguito a errori di istruzione impostata nella configurazione del PLC	Il valore specificato in un indirizzo DM/EM indiretto in modalità BCD non è in formato BCD.	Se nella configurazione del PLC è impostata l'interruzione del funzionamento in seguito a errori di istruzione, vengono attivati il flag AER e il flag di errore BCD DM/EM indiretto (A29509).
Errore di overflow comparativo degli indirizzi	Durante la modifica in linea, sono state inserite o eliminate più di 131.071 istruzioni differenziate.	Viene attivato il flag di errore di overflow comparativo (A29513).
Errore di overflow UM (memoria utente)	Si è tentato di eseguire dati di istruzioni memorizzati in indirizzi successivi all'ultimo indirizzo della memoria utente (UM, User Memory) definita come area di archiviazione del programma.	Viene attivato il flag di overflow UM (A29516).
Errore di istruzione illegale	Si è tentato di eseguire un'istruzione che non può essere eseguita.	Viene attivato il flag di istruzione illegale (A29514).

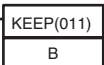
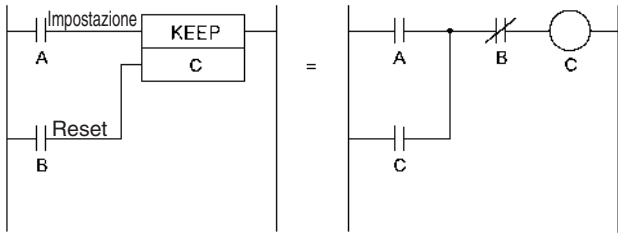
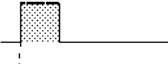
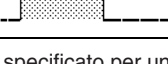
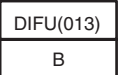
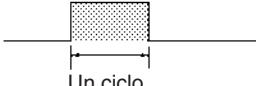
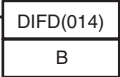
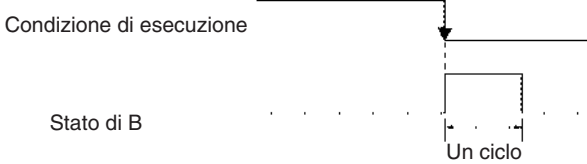
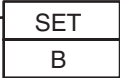
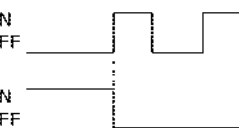
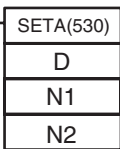
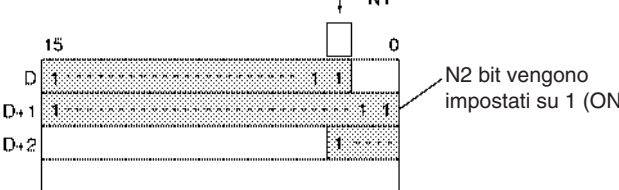
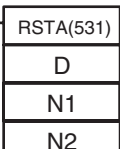
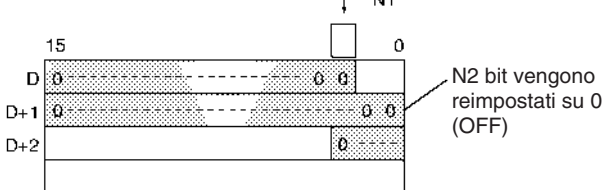
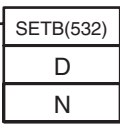
CAPITOLO 3
Funzioni di istruzione

In questo capitolo viene fornita una descrizione generale delle istruzioni che è possibile utilizzare per la creazione di programmi utente.

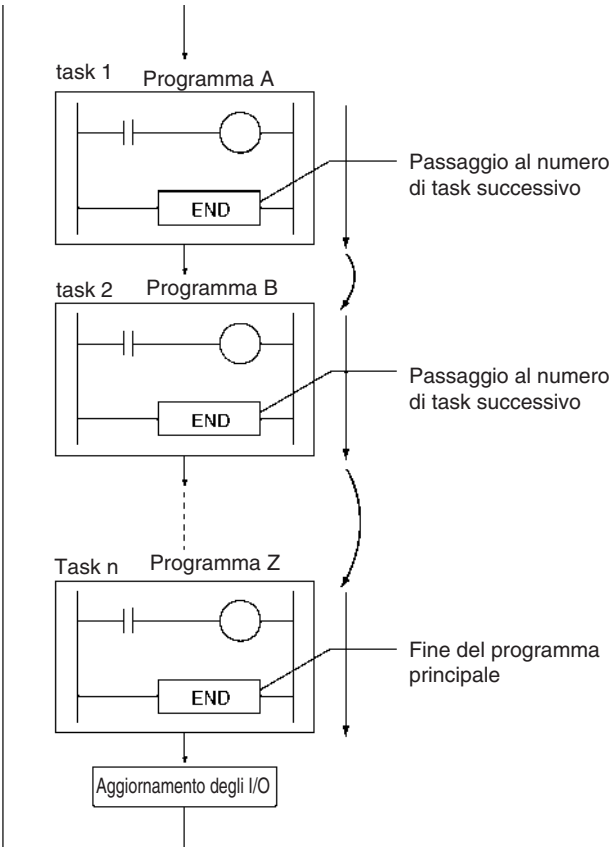
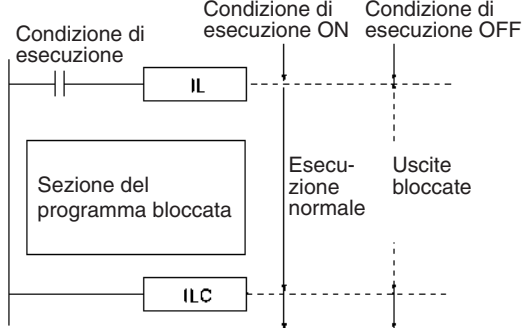
3-1 Istruzioni di ingresso sequenza

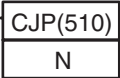
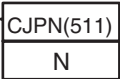
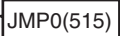
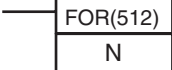
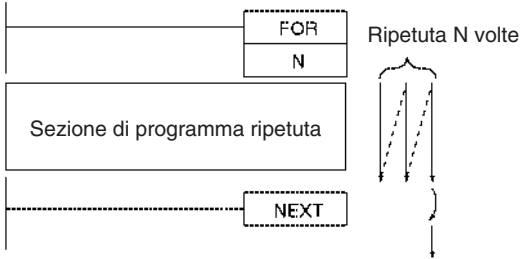
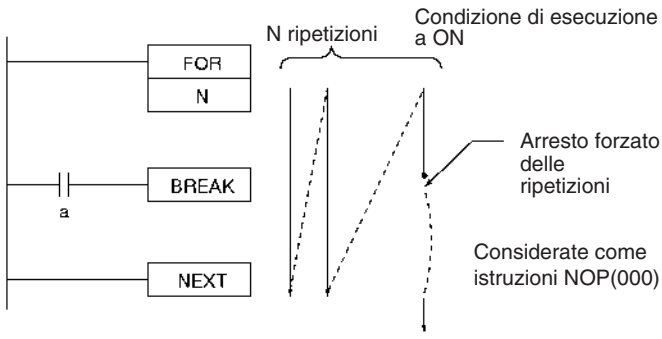
*1: non supportata dalle CPU CS1D per sistemi a due CPU.

*2: supportata solo dalle CPU CS1-H, CJ1-H, CJ1M e CS1D.

*3: supportata solo dalle CPU CS1-H, CJ1-H e CJ1M.

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
LOAD LD @LD %LD !LD*1 !@LD*1 !%LD*1	Barra di distribuzione Punto iniziale del blocco	Indica un inizio logico e crea una condizione di esecuzione ON/OFF basata sullo stato ON/OFF del bit operando specificato.	Inizio della logica Non richiesta
LOAD NOT LD NOT @LD NOT*2 %LD NOT*2 !LD NOT*1 !@LD NOT*3 !%LD NOT*3	Barra di distribu-zione Punto iniziale del blocco	Indica un inizio logico e crea una condizione di esecuzione ON/OFF basata sull'inverso dello stato ON/OFF del bit operando specificato.	Inizio della logica Non richiesta
AND AND @AND %AND !AND*1 !@AND*1 !%AND*1		Esegue un AND logico dello stato del bit operando specificato e della condizione di esecuzione corrente.	Continua su rung Richiesta
AND NOT AND NOT @AND NOT*2 %AND NOT*2 !AND NOT*1 !@AND NOT*3 !%AND NOT*3		Inverte lo stato del bit operando specificato ed esegue un AND logico con la condizione di esecuzione corrente.	Continua su rung Richiesta
OR OR @OR %OR !OR*1 !@OR*1 !%OR*1	Barra di distribuzione	Esegue un OR logico dello stato ON/OFF del bit operando specificato e della condizione di esecuzione corrente.	Continua su rung Richiesta
OR NOT OR NOT @OR NOT*2 %OR NOT*2 !OR NOT*1 !@OR NOT*3 !%OR NOT*3	Barra di distribuzione	Inverte lo stato del bit specificato ed esegue un OR logico con la condizione di esecuzione corrente.	Continua su rung Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
AND LOAD AND LD	Blocco logico - Blocco logico	Esegue un AND logico tra blocchi logici. LD a } Blocco logico A LD a } Blocco logico B AND LD .... Collegamento seriale tra i blocchi logici A e B.	Continua su rung Richiesta
OR LOAD OR LD	Blocco logico Blocco logico	Esegue un OR logico tra blocchi logici. LD a } Blocco logico A LD a } Blocco logico B OR LD .... Collegamento parallelo tra i blocchi logici A e B.	Continua su rung Richiesta
NOT NOT 520	---	Inverte la condizione di esecuzione.	Continua su rung Richiesta
CONDITION ON UP 521	UP(521)	UP(521) attiva la condizione di esecuzione per un ciclo quando lo stato di quest'ultima passa da OFF a ON.	Continua su rung Richiesta
CONDITION OFF DOWN 522	DOWN(522)	DOWN(522) attiva la condizione di esecuzione per un ciclo quando lo stato di quest'ultima passa da ON a OFF.	Continua su rung Richiesta
BIT TEST LD TST 350	TST(350) S N **S:** canale sorgente **N:** numero bit	LD TST(350), AND TST(350) e OR TST(350) vengono utilizzate nel programma analogamente a LD, AND e OR. La condizione di esecuzione viene attivata quando il bit specificato nel canale specificato viene impostato su ON e disattivata quando il bit viene impostato su OFF.	Continua su rung Non richiesta
BIT TEST LD TSTN 351	TSTN(351) S N **S:** canale sorgente **N:** numero bit	LD TSTN(351), AND TSTN(351) e OR TSTN(351) vengono utilizzate nel programma analogamente a LD NOT, AND NOT e OR NOT. La condizione di esecuzione viene disattivata quando il bit specificato nel canale specificato viene impostato su ON e attivata quando il bit viene impostato su OFF.	Continua su rung Non richiesta
BIT TEST AND TST 350	AND TST(350) S N **S:** canale sorgente **N:** numero bit	LD TST(350), AND TST(350) e OR TST(350) vengono utilizzate nel programma analogamente a LD, AND e OR. La condizione di esecuzione viene attivata quando il bit specificato nel canale specificato viene impostato su ON e disattivata quando il bit viene impostato su OFF.	Continua su rung Richiesta
BIT TEST AND TSTN 351	AND TSTN(351) S N **S:** canale sorgente **N:** numero bit	LD TSTN(351), AND TSTN(351) e OR TSTN(351) vengono utilizzate nel programma analogamente a LD NOT, AND NOT e OR NOT. La condizione di esecuzione viene disattivata quando il bit specificato nel canale specificato viene impostato su ON e attivata quando il bit viene impostato su OFF.	Continua su rung Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
BIT TEST OR TST 350	TST(350) S N **S:** canale sorgente **N:** numero bit	LD TST(350), AND TST(350) e OR TST(350) vengono utilizzate nel programma analogamente a LD, AND e OR. La condizione di esecuzione viene attivata quando il bit specificato nel canale specificato viene impostato su ON e disattivata quando il bit viene impostato su OFF.	Continua su rung Richiesta
BIT TEST OR TSTN 351	TSTN(351) S N **S:** canale sorgente **N:** numero bit	LD TSTN(351), AND TSTN(351) e OR TSTN(351) vengono utilizzate nel programma analogamente a LD NOT, AND NOT e OR NOT. La condizione di esecuzione viene disattivata quando il bit specificato nel canale specificato viene impostato su ON e attivata quando il bit viene impostato su OFF.	Continua su rung Richiesta

3-2 Istruzioni di uscita sequenza

*1: non supportata dalle CPU CS1D per sistemi a due CPU.

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
OUTPUT OUT !OUT*1		Invia in uscita il risultato (condizione di esecuzione) dell'elaborazione logica al bit specificato.	Uscita Richiesta
OUTPUT NOT OUT NOT !OUT NOT*1		Inverte il risultato (condizione di esecuzione) dell'elaborazione logica e lo invia in uscita al bit specificato.	Uscita Richiesta
KEEP KEEP !KEEP*1 011	S (impostazione) KEEP(011) B R (reset) **B:** Bit	Funziona come un relé di blocco. Impostazione A, Reset B, KEEP C = A, B, C Condizione di esecuzione S Condizione di esecuzione R Stato di B	Uscita Richiesta
DIFFERENTIATE UP DIFU !DIFU*1 013	DIFU(013) B **B:** Bit	DIFU(013) imposta su ON il bit specificato per un ciclo quando la condizione di esecuzione passa da OFF a ON (fronte di salita). Condizione di esecuzione Stato di B Un ciclo	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DIFFERENTIATE DOWN DIFD !DIFD*1 014	DIFD(014) B **B:** Bit	DIFD(014) imposta su ON il bit specificato per un ciclo quando la condizione di esecuzione passa da ON a OFF (fronte di discesa). Condizione di esecuzione Stato di B Un ciclo	Uscita Richiesta
SET SET @SET %SET !SET*1 !@SET*1 !%SET*1	SET B **B:** Bit	SET imposta il bit dell'operando su ON quando la condizione di esecuzione è ON. Condizione di esecuzione di SET ON OFF Stato di B ON OFF	Uscita Richiesta
RESET RSET @RSET %RSET !RSET*1 !@RSET*1 !%RSET*1	RSET B **B:** Bit	RSET imposta il bit dell'operando su OFF quando la condizione di esecuzione è ON. Condizione di esecuzione di RSET ON OFF Stato di B ON OFF	Uscita Richiesta
MULTIPLE BIT SET SETA @SETA 530	SETA(530) D N1 N2 **D:** canale iniziale **N1:** bit iniziale **N2:** numero di bit	SETA(530) imposta su ON il numero di bit consecutivi specificato. N1 15 0 D 1 ... 1 1 D+1 1 ... 1 1 D+2 1 ... N2 bit vengono impostati su 1 (ON)	Uscita Richiesta
MULTIPLE BIT RESET RSTA @RSTA 531	RSTA(531) D N1 N2 **D:** canale iniziale **N1:** bit iniziale **N2:** numero di bit	RSTA(531) imposta su OFF il numero di bit consecutivi specificato. N1 15 0 D 0 ... 0 0 D+1 0 ... 0 0 D+2 0 ... N2 bit vengono reimpostati su 0 (OFF)	Uscita Richiesta
SINGLE BIT SET (solo CS1-H, CJ1-H, CJ1M o CS1D) SETB @SETB !SETB*1 !@SETB*1	SETB(532) D N **D:** indirizzo del canale **N:** numero bit	SETB(532) imposta su ON il bit specificato nel canale specificato quando lo stato della condizione di esecuzione è ON. A differenza dell'istruzione SET, è possibile utilizzare SETB(532) per impostare un bit in un canale DM o EM.	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
SINGLE BIT RESET (solo CS1-H, CJ1-H, CJ1M o CS1D) RSTB @RSTB !RSTB*1 !@RSTB*1	RSTB(533) D N **D:** indirizzo del canale **N:** numero bit	RSTB(533) imposta su OFF il bit specificato nel canale specificato quando lo stato della condizione di esecuzione è ON. A differenza dell'istruzione RSET, è possibile utilizzare RSTB(533) per reimpostare un bit in un canale DM o EM.	Uscita Richiesta
SINGLE BIT OUTPUT (solo CS1-H, CJ1-H, CJ1M o CS1D) OUTB @OUTB !OUTB*1	OUTB(534) D N **D:** indirizzo del canale **N:** numero bit	OUTB(534) invia in uscita il risultato (condizione di esecuzione) dell'elaborazione logica al bit specificato. A differenza dell'istruzione OUT, è possibile utilizzare OUTB(534) per controllare un bit di un canale DM o EM.	Uscita Richiesta

3-3 Istruzioni di controllo sequenza

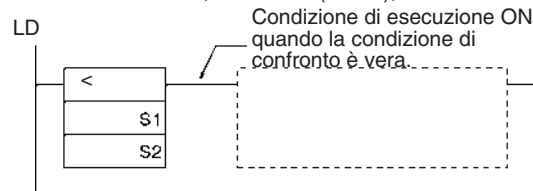
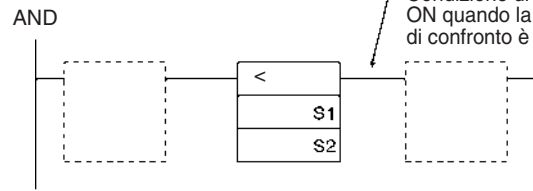
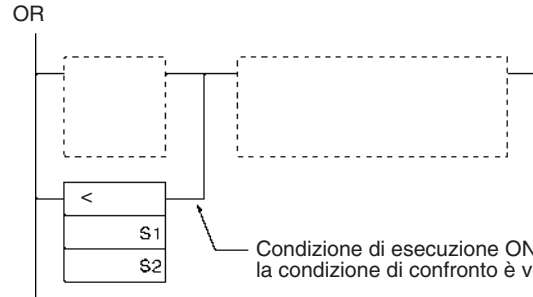
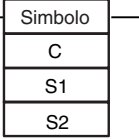
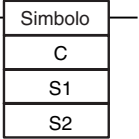
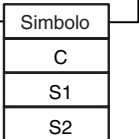
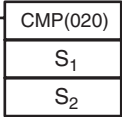
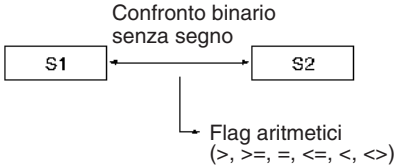
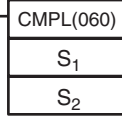
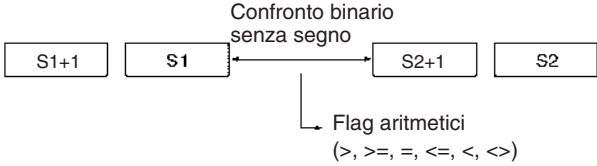
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
END END 001	END (001)	Indica la fine del programma. END(001) termina l'esecuzione di un programma in quel ciclo. Le istruzioni inserite dopo END(001) non vengono eseguite. Viene eseguito il programma con il numero di task successivo. Se numero di task del programma in esecuzione è il più alto all'interno del programma, l'istruzione END(001) marca la fine del programma principale. task 1 Programma A; END; Passaggio al numero di task successivo task 2 Programma B; END; Passaggio al numero di task successivo Task n Programma Z; END; Fine del programma principale Aggiornamento degli I/O	Uscita Non richiesta
NO OPERATION NOP 000		Istruzione priva di funzione. Per NOP(000) non viene eseguita alcuna elaborazione.	Uscita Non richiesta
INTERLOCK IL 002	IL(002)	Determina l'interblocco per tutte le uscite comprese tra IL(002) e ILC(003) quando lo stato della condizione di esecuzione per IL(002) è OFF. IL(002) e ILC(003) vengono di norma utilizzate in coppia. Condizione di esecuzione; IL; Condizione di esecuzione ON; Condizione di esecuzione OFF; Sezione del programma bloccata; Esecu- zione normale; Uscite bloccate; ILC	Uscita Richiesta

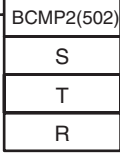
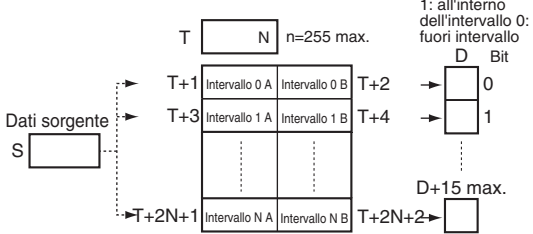
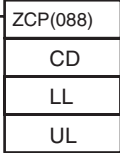
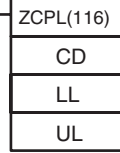
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
INTERLOCK CLEAR ILC 003	ILC(003)	Interblocco di tutte le uscite comprese tra le istruzioni IL(002) e ILC(003) quando la condizione di esecuzione per IL(002) è OFF. IL(002) e ILC(003) vengono di norma utilizzate in coppia.	Uscita Non richiesta
MULTI-INTERLOCK DIFFERENTIATIO N HOLD MILH 517 CPU della serie CS/CJ versione 2.0 o successiva	MILH(517) N D **N:** numero di interblocco **D:** bit di stato interblocco	Quando la condizione di esecuzione per MILH(517) è OFF, le uscite di tutte le istruzioni comprese tra tale istruzione MILH(517) e l'istruzione MILC(519) successiva vengono interbloccate. Le istruzioni MILH(517) e MILC(519) vengono utilizzate in coppia. Gli interblocchi MILH(517)/MILC(519) possono essere annidati [ad esempio MILH(517)—MILH(517)—MILC(519)—MILC(519)]. Se tra l'istruzione MILH(517) e l'istruzione MILC(519) corrispondente è presente un'istruzione differenziata (DIFU, DIFD o un'istruzione con prefisso @ o %), tale istruzione verrà eseguita dopo la cancellazione dell'interblocco se viene soddisfatta la condizione di differenziazione dell'istruzione.	Uscita Richiesta
MULTI-INTERLOCK DIFFERENTIATIO N RELEASE MILR 518 CPU della serie CS/CJ versione 2.0 o successiva	MILR(518) N D **N:** numero di interblocco **D:** bit di stato interblocco	Quando la condizione di esecuzione per MILR(518) è OFF, le uscite di tutte le istruzioni comprese tra tale istruzione MILR(518) e l'istruzione MILC(519) successiva vengono interbloccate. Le istruzioni MILR(518) e MILC(519) vengono utilizzate in coppia. Gli interblocchi MILR(518)/MILC(519) possono essere annidati [ad esempio MILR(518)—MILR(518)—MILC(519)—MILC(519)]. Se tra l'istruzione MILR(518) e l'istruzione MILC(519) corrispondente è presente un'istruzione differenziata (DIFU, DIFD o un'istruzione con prefisso @ o %), tale istruzione non verrà eseguita dopo la cancellazione dell'interblocco anche se viene soddisfatta la condizione di differenziazione dell'istruzione.	Uscita Richiesta
MULTI-INTERLOCK CLEAR MILC 519 CPU della serie CS/CJ versione 2.0 o successiva	MILC(519) N **N:** numero di interblocco	Cancella l'interblocco avviato da un'istruzione MILH(517) o MILR(518) con il medesimo numero di interblocco. Tutte le uscite comprese tra l'istruzione MILH(517)/MILR(518) e l'istruzione MILC(519) corrispondente con lo stesso numero di interblocco vengono interbloccate quando la condizione di esecuzione per MILH(517)/MILR(518) è OFF.	Uscita Non richiesta
JUMP JMP 004	JMP(004) N **N:** numero di salto	Quando la condizione di esecuzione per JMP(004) è OFF, l'esecuzione del programma passa direttamente alla prima istruzione JME(05) nel programma avente lo stesso numero di salto. Le istruzioni JMP(004) e JME(005) vengono utilizzate in coppia. Condizione di esecuzione JMP N ON OFF Istruzioni saltate Istruzioni eseguite Le istruzioni in questa sezione non vengono eseguite e lo stato di uscita viene mantenuto. Il tempo di esecuzione di tali istruzioni è pertanto inesistente. JME N	Uscita Richiesta
JUMP END JME 005	JME(005) N **N:** numero di salto	Indica la fine di un salto il cui inizio è specificato dall'istruzione JMP(004) or CJP(510).	Uscita Non richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
CONDITIONAL JUMP NOT CJP 510	CJP(510) N **N:** numero di salto	Il funzionamento di CJP(510) è sostanzialmente opposto a quello di JMP(004). Quando la condizione di esecuzione per CJP(510) è ON, l'esecuzione del programma passa direttamente alla prima istruzione JME(05) nel programma avente lo stesso numero di salto. Le istruzioni CJP(510) e JME(005) vengono utilizzate in coppia. Condizione di esecuzione OFF / Condizione di esecuzione ON CJP N Istruzioni eseguite JME N Istruzioni saltate Le istruzioni in questa sezione non vengono eseguite e lo stato di uscita viene mantenuto. Il tempo di esecuzione di tali istruzioni è pertanto inesistente.	Uscita Richiesta
CONDITIONAL JUMP NOT CJPN 511	CJPN(511) N **N:** numero di salto	Il funzionamento di CJPN(511) è quasi uguale a quello di JMP(004). Quando la condizione di esecuzione per CJP(004) è OFF, l'esecuzione del programma passa direttamente alla prima istruzione JME(05) nel programma avente lo stesso numero di salto. Le istruzioni CJPN(511) e JME(005) vengono utilizzate in coppia. Condizione di esecuzione ON / Condizione di esecuzione OFF CJPN N Istruzioni eseguite JME N Istruzioni saltate Le istruzioni in questa sezione non vengono eseguite e lo stato di uscita viene mantenuto. Il tempo di esecuzione di tali istruzioni è pertanto inesistente.	Uscita Non richiesta
MULTIPLE JUMP JMP0 515	JMP0(515)	Quando lo stato della condizione di esecuzione per JMP0(515) è OFF, tutte le istruzioni del programma comprese tra JMP0(515) e l'istruzione JME0(516) successiva vengono elaborate come NOP(000). Utilizzare JMP0(515) e JME0(516) in coppia. Nel programma è possibile utilizzare un numero illimitato di istruzioni JMP0(515)/JME0(516) in coppia. Condizione di esecuzione a ON / Condizione di esecuzione a OFF a JMP0 Istruzioni eseguite JME0 Istruzioni saltate Condizione di esecuzione b ON / Condizione di esecuzione b OFF b JMP0 Istruzioni eseguite JME0 Istruzioni saltate Le istruzioni saltate vengono considerate come istruzioni NOP(000) e richiedono lo stesso tempo di esecuzione.	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
MULTIPLE JUMP END JME0 516	JME0(516)	Quando lo stato della condizione di esecuzione per JMP0(515) è OFF, tutte le istruzioni del programma comprese tra JMP0(515) e l'istruzione JME0(516) successiva vengono elaborate come NOP(000). Utilizzare JMP0(515) e JME0(516) in coppia. Nel programma è possibile utilizzare un numero illimitato di istruzioni JMP0(515)/JME0(516) in coppia.	Uscita Non richiesta
FOR-NEXT LOOPS FOR 512	FOR(512) N **N:** numero di cicli	Le istruzioni comprese tra FOR(512) e NEXT(513) vengono ripetute per il numero di volte specificato. Le istruzioni FOR(512) e NEXT(513) devono essere utilizzate in coppia. FOR N Ripetuta N volte Sezione di programma ripetuta NEXT	Uscita Non richiesta
BREAK LOOP BREAK 514	BREAK(514)	Viene inserita in un ciclo FOR-NEXT per interrompere l'esecuzione del ciclo quando si verifica una determinata condizione. Le rimanenti istruzioni all'interno del ciclo vengono considerate come istruzioni NOP(000) N ripetizioni Condizione di esecuzione a ON FOR N Arresto forzato delle ripetizioni BREAK a Considerate come istruzioni NOP(000) NEXT	Uscita Richiesta
FOR-NEXT LOOPS NEXT 513	NEXT(513)	Le istruzioni comprese tra FOR(512) e NEXT(513) vengono ripetute per il numero di volte specificato. Le istruzioni FOR(512) e NEXT(513) devono essere utilizzate in coppia.	Uscita Non richiesta

3-4 Istruzioni di temporizzatore e contatore

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
TIMER TIM (BCD) TIMX (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	TIM N S **N:** numero di temporizzatore **S:** Valore impostato TIMX(550) N S **N:** numero di temporizzatore **S:** Valore impostato	TIM/TIMX(550) aziona un temporizzatore decrementale con unità di 0,1 s. La gamma di impostazione per il valore impostato (SV) è compresa tra 0 e 999,9 s per il BCD e tra 0 e 6.553,5 s per il formato binario (decimale o esadecimale). Ingresso del temporizzatore ON OFF PV del temporizzatore SV 0 Completamento Flag ON OFF Ingresso del temporizzatore ON OFF PV del temporizzatore SV 0 Completamento Flag ON OFF	Uscita Richiesta
HIGH-SPEED TIMER TIMH 015 (BCD) TIMHX 551 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	TIMH(015) N S **N:** numero di temporizzatore **S:** Valore impostato TIMHX(551) N S **N:** numero di temporizzatore **S:** Valore impostato	TIMH(015)/TIMHX(551) aziona un temporizzatore decrementale con unità di 10 ms. La gamma di impostazione per il valore impostato (SV) è compresa tra 0 e 99,99 s per il formato BCD e tra 0 e 655,35 s per il formato binario (decimale o esadecimale). Ingresso del temporizzatore ON OFF PV del temporizzatore SV 0 Completamento Flag ON OFF Ingresso del temporizzatore ON OFF PV del temporizzatore SV 0 Completamento Flag ON OFF	Uscita Richiesta
ONE-MS TIMER TMHH 540 (BCD) TMHHX 552 (BCD) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	TMHH(540) N S **N:** numero di temporizzatore **S:** Valore impostato TMHHX(552) N S **N:** numero di temporizzatore **S:** Valore impostato	TMHH(540)/TMHHX(552) aziona un temporizzatore decrementale con unità di 1 ms. La gamma di impostazione per il valore impostato (SV) è compresa tra 0 e 9,999 s per il formato BCD e tra 0 e 65,535 s per il formato binario (decimale o esadecimale). I diagrammi di tempistica per TMHH(540) e TMHHX(552) sono uguali a quelli indicati in precedenza per TIMH(015).	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
ACCUMULATIVE TIMER TTIM 087 (BCD) TTIMX 555 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	Ingresso del temporizzatore — TTIM(087) N S; Ingresso di reset **N:** numero di temporizzatore **S:** Valore impostato Ingresso del temporizzatore — TTIMX(555) N S; Ingresso di reset **N:** numero di temporizzatore **S:** Valore impostato	TTIM(087)/TTIMX(555) aziona un temporizzatore incrementale con unità di 0,1 s. La gamma di impostazione per il valore impostato (SV) è compresa tra 0 e 999,9 s per il formato BCD e tra 0 e 6.553,5 s per il formato binario (decimale o esadecimale).	Uscita Richiesta
LONG TIMER TIML 542 (BCD) TIMLX 553 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	TIML(542) D1 D2 S **D1:** Flag di completamento **D2:** canale PV **S:** canale SV TIMLX(553) D1 D2 S **D1:** Flag di completamento **D2:** canale PV **S:** canale SV	TIML(542)/TIMLX(553) aziona un temporizzatore decrementale con unità di 0,1 s che può raggiungere un periodo di circa 115 giorni per il formato BCD e 49.710 giorni per il formato binario (decimale o esadecimale).	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
MULTI-OUTPUT TIMER MTIM 543 (BCD) MTIMX 554 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	MTIM(543) D1 D2 S **D1:** flag di completamento **D2:** canale PV **S:** primo canale SV MTIMX(554) D1 D2 S **D1:** flag di completamento **D2:** canale PV **S:** primo canale SV	MTIM(543)/MTIMX(554) aziona un temporizzatore incrementale con unità di 0,1 s che dispone di 8 flag di completamento e valori impostati indipendenti. La gamma di impostazione per il valore impostato (SV) è compresa tra 0 e 999,9 secondi per valori decimali codificati in binario (BCD) e da 0 a 6.553,5 secondi per valori binari (decimali o esadecimali).	Uscita Richiesta
COUNTER CNT (BCD) CNTX 546 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	Ingresso di conteggio, Ingresso di reset CNT N S **N:** Numero contatore **S:** Valore impostato Ingresso di conteggio, Ingresso di reset CNTX(546) N S **N:** numero contatore **S:** Valore impostato	CNT/CNTX(546) attiva un contatore decrementale. La gamma di impostazione per il valore impostato (SV) è compresa tra 0 e 9.999 secondi per valori decimali codificati in binario (BCD) e da 0 a 65.535 secondi per valori binari (decimali o esadecimali).	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
REVERSIBLE COUNTER CNTR 012 (BCD) CNTRX 548 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	Ingresso di incremento, Ingresso di decremento, Ingresso di reset — CNTR(012) N S **N:** Numero contatore **S:** Valore impostato Ingresso di incremento, Ingresso di decremento, Ingresso di reset — CNTRX(548) N S **N:** numero contatore **S:** valore impostato	CNTR(012)/CNTRX(548) attiva un contatore reversibile. Ingresso di incremento Ingresso di decremento PV del contatore 0 PV del contatore SV 0 +1 Flag di completamento ON OFF PV del contatore SV 0 1 Flag di completamento ON OFF	Uscita Richiesta
RESET TIMER/ COUNTER CNR @CNR 545 (BCD) CNRX @CNRX 547 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	CNR(545) N1 N2 **N1:** primo numero nell'intervallo **N2:** ultimo numero nell'intervallo CNRX(547) N1 N2 **N1:** primo numero nell'intervallo **N2:** ultimo numero nell'intervallo	CNR(545)/CNRX(547) reimposta i temporizzatori o i contatori che rientrano nell'intervallo dei numeri di temporizzatore o contatore specificato. Imposta il valore impostato su 9999 (valore massimo).	Uscita Richiesta

3-5 Istruzioni di confronto

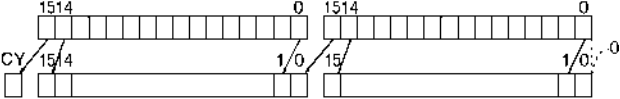
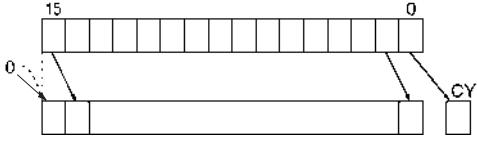
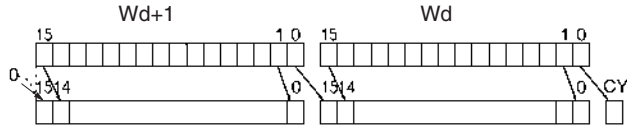
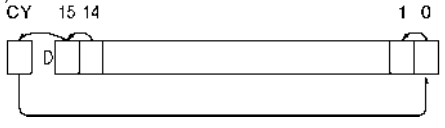
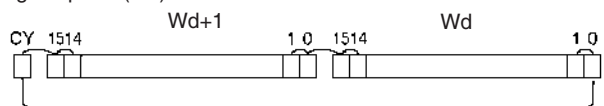
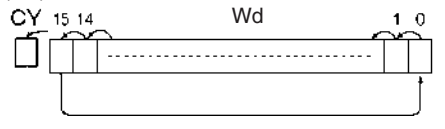
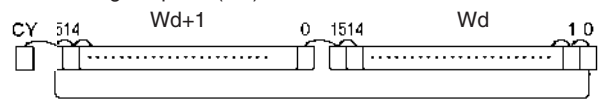
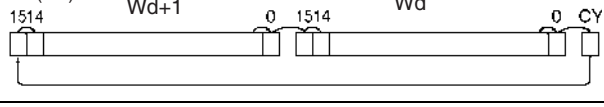
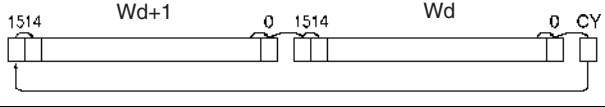
*1: non supportata dalle CPU CS1D per sistemi a due CPU.

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
Confronto con simboli (senza segno) LD, AND, OR + =, <>, <, <=, >, >= 300 (=) 305 (<>) 310 (<) 315 (<=) 320 (>) 325(>=)	Simbolo e opzioni S_1 S_2 **S_1:** dati di confronto 1 **S_2:** dati di confronto 2	Le istruzioni di confronto con simboli (senza segno) confrontano due valori (costanti e/o il contenuto di canali specifici) in formato binario a 16 bit e creano una condizione di esecuzione ON quando la condizione di confronto è vera. Esistono tre tipi di istruzioni di confronto con simboli, ovvero LD (LOAD), AND e OR. LD Condizione di esecuzione ON quando la condizione di confronto è vera. < S1 S2 AND Condizione di esecuzione ON quando la condizione di confronto è vera. < S1 S2 OR < S1 S2 Condizione di esecuzione ON quando la condizione di confronto è vera.	LD: non richiesta AND, OR: richiesta
Confronto con simboli (doppio canale, senza segno) LD, AND, OR + =, <>, <, <=, >, >= + L 301 (=) 306 (<>) 311 (<) 316 (<=) 321 (>) 326 (>=)	**S_1**: dati di confronto 1 **S_2**: dati di confronto 2	Le istruzioni di confronto con simboli (doppio canale, senza segno) confrontano due valori (costanti e/o il contenuto di dati specifici a doppio canale) in formato binario a 32 bit senza segno e creano una condizione di esecuzione ON quando la condizione di confronto è vera. Esistono tre tipi di istruzioni di confronto con simboli, ovvero LD (LOAD), AND e OR.	LD: non richiesta AND, OR: richiesta
Confronto con simboli (con segno) LD, AND, OR + =, <>, <, <=, >, >= +S 302 (=) 307 (<>) 312 (<) 317 (<=) 322 (>) 327 (>=)	**S_1**: dati di confronto 1 **S_2**: dati di confronto 2	Le istruzioni di confronto con simboli (con segno) confrontano due valori (costanti e/o il contenuto di canali specifici) in formato binario a 16 bit con segno (esadecimale a 4 cifre) e creano una condizione di esecuzione ON quando la condizione di confronto è vera. Esistono tre tipi di istruzioni di confronto con simboli, ovvero LD (LOAD), AND e OR.	LD: non richiesta AND, OR: richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
Confronto con simboli (doppio canale, con segno) LD, AND, OR + =, <>, <, <=, >, >= +SL 303 (=) 308 (<>) 313 (<) 318 (<=) 323 (>) 328 (>=)	**S_1**: dati di confronto 1 **S_2**: dati di confronto 2	Le istruzioni di confronto con simboli (doppio canale, con segno) confrontano due valori (costanti e/o il contenuto di dati specifici a doppio canale) in formato binario con segno a 32-bit (esadecimale a 8 cifre) e creano una condizione di esecuzione ON quando la condizione di confronto è vera. Esistono tre tipi di istruzioni di confronto con simboli, ovvero LD (LOAD), AND e OR.	LD: Non richiesta AND, OR: richiesta
Confronto di date e ore LD, AND, OR + = DT, <> DT, < DT, <= DT, > DT, >= DT 341 (= DT) 342 (<> DT) 343 (< DT) 344 (<= DT) 345 (> DT) 346 (>= DT) (solo CPU della serie CS/CJ versione 2.0 o successiva)	LD (LOAD): Simbolo C S1 S2 AND: Simbolo C S1 S2 OR: Simbolo C S1 S2 **C:** canale di controllo **S1:** primo canale del tempo attuale **S2:** primo canale del tempo di confronto	Le istruzioni di confronto di date e ore confrontano due valori temporali in formato BCD e creano una condizione di esecuzione ON quando la condizione di confronto è soddisfatta. Esistono tre tipi di istruzioni di confronto di date e ore, ovvero LD (LOAD), AND e OR. I valori (anno, mese, giorno, ora, minuto e secondo) possono essere mascherati/smascherati nel confronto per semplificare la creazione di funzioni di temporizzatore calendario.	LD: Non richiesta AND, OR: richiesta
UNSIGNED COMPARE CMP !CMP*1 020	CMP(020) S_1 S_2 **S1:** dati di confronto 1 **S2:** dati di confronto 2	Confronta due valori binari senza segno (costanti e/o il contenuto di canali specifici) e invia il risultato ai flag aritmetici dell'area ausiliaria. Confronto binario senza segno S1 ↔ S2 Flag aritmetici (>, >=, =, <=, <, <>)	Uscita Richiesta
DOUBLE UNSIGNED COMPARE CMPL 060	CMPL(060) S_1 S_2 **S_1:** dati di confronto 1 **S_2:** dati di confronto 2	Confronta due valori binari senza segno doppi (costanti e/o il contenuto di canali specifici) e invia il risultato ai flag aritmetici dell'area ausiliaria. Confronto binario senza segno S1+1 S1 ↔ S2+1 S2 Flag aritmetici (>, >=, =, <=, <, <>)	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SIGNED BINARY COMPARE CPS !CPS*1 114	CPS(114) S_1 S_2 **S1:** dati di confronto 1 **S2:** dati di confronto 2	Confronta due valori binari con segno (costanti e/o il contenuto di canali specifici) e invia il risultato ai flag aritmetici dell'area ausiliaria. Confronto binario con segno S1 S2 Flag aritmetici (>, >=, =, <=, <, <>)	Uscita Richiesta
DOUBLE SIGNED BINARY COMPARE CPSL 115	CPSL(115) S_1 S_2 **S1:** dati di confronto 1 **S2:** dati di confronto 2	Confronta due valori binari con segno doppi (costanti e/o il contenuto di canali specifici) e invia il risultato ai flag aritmetici dell'area ausiliaria. Confronto binario con segno S1+1 S1 S2+1 S2 Flag aritmetici (>, >=, =, <=, <, <>)	Uscita Richiesta
MULTIPLE COMPARE MCMP @MCMP 019	MCMP(019) S_1 S_2 R **S1:** primo canale del gruppo 1 **S2:** primo canale del gruppo 2 **R:** canale del risultato	Confronta 16 canali consecutivi con altri 16 canali consecutivi e, nel caso in cui il contenuto di un canale **non è** uguale, imposta su ON il bit corrispondente nel canale del risultato. Confronto S1, S1+1, S1+14, S1+15 — S2, S2+1, S2+14, S2+15 — R: 0, 1, 14, 15 0: i canali sono uguali. 1: i canali non sono uguali.	Uscita Richiesta
TABLE COMPARE TCMP @TCMP 085	TCMP(085) S T R **S:** Dati sorgente **T:** primo canale della tabella **R:** canale del risultato	Confronta i dati sorgente con il contenuto di 16 canali e, se il contenuto è uguale, imposta su ON il bit corrispondente nel canale del risultato. Confronto S — T, T+1, T+14, T+15 — R: 0, 1, 14, 15 1: i dati sono uguali. 0: i dati non sono uguali.	Uscita Richiesta
UNSIGNED BLOCK COMPARE BCMP @BCMP 068	BCMP(068) S T R **S:** Dati sorgente **T:** primo canale della tabella **R:** canale del risultato	Confronta i dati sorgente con 16 intervalli (definiti da 16 limiti inferiori e 16 limiti superiori) e, se i dati sorgente rientrano nell'intervallo, imposta su ON il bit corrispondente nel canale del risultato. Intervalli 1: all'interno dell'intervallo 0: fuori intervallo Limite inferiore Limite superiore R Dati sorgente S T a T+1 → 0 T+2 a T+3 → 1 Da T+28 a T+29 → 14 Da T+30 a T+31 → 15	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
EXPANDED BLOCK COMPARE BCMP2 @BCMP2 502 (solo CPU CS1-H, CJ1-H o CS1D versione 2.0 o successiva) CPU CJ1M (precedenti alla versione 2.0 o versione 2.0 o successiva)	BCMP2(502) S T R **S:** Dati sorgente **T:** primo canale del blocco **R:** canale del risultato	Confronta i dati di origine con intervalli definiti da un limite inferiore e superiore, fino a un massimo di 256, e quando i dati di origine rientrano in un intervallo, imposta su ON il bit corrispondente nel canale del risultato. T: N n=255 max.; T+1: Intervallo 0 A, T+2: Intervallo 0 B; T+3: Intervallo 1 A, T+4: Intervallo 1 B; T+2N+1: Intervallo N A, T+2N+2: Intervallo N B. Dati sorgente S. D Bit 0, 1 … D+15 max. (1: all'interno dell'intervallo 0: fuori intervallo) **Nota:** A può essere inferiore, uguale o superiore a B.	Uscita Richiesta
AREA RANGE COMPARE ZCP @ZCP 088 (solo CS1-H, CJ1-H, CJ1M o CS1D)	ZCP(088) CD LL UL **CD:** dati da confrontare (1 canale) **LL:** limite inferiore dell'intervallo **UL:** limite superiore dell'intervallo	Confronta il valore binario a 16 bit senza segno in CD (contenuto del canale o costante) con l'intervallo definito da LL e UL e invia i risultati ai flag aritmetici dell'area ausiliaria.	Uscita Richiesta
DOUBLE AREA RANGE COMPARE ZCPL @ZCPL 116 (solo CS1-H, CJ1-H, CJ1M o CS1D)	ZCPL(116) CD LL UL **CD:** dati da confrontare (2 canali) **LL:** limite inferiore dell'intervallo **UL:** limite superiore dell'intervallo	Confronta il valore binario a 32 bit senza segno in CD e CD+1 (contenuto del canale o costante) con l'intervallo definito da LL e UL e invia i risultati ai flag aritmetici dell'area ausiliaria.	Uscita Richiesta

3-6 Istruzioni di spostamento dei dati

Istruzione Codice mnemonico Codice	Simbolo/operando	Funzione	Posizione Condizione di esecuzione
MOVE MOV @MOV !MOV !@MOV 021	MOV(021) S D **S:** sorgente **D:** destinazione	Trasferisce un canale di dati nel canale specificato. canale sorgente Stato del bit non modificato Canale di destinazione	Uscita Richiesta
DOUBLE MOVE MOVL @MOVL 498	MOVL(498) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Trasferisce due canali di dati nei canali specificati. S S+1 Stato del bit non modificato D D+1	Uscita Richiesta
MOVE NOT MVN @MVN 022	MVN(022) S D **S:** sorgente **D:** destinazione	Trasferisce il complemento di un canale di dati nel canale specificato. canale sorgente Stato del bit invertito Canale di destinazione	Uscita Richiesta
DOUBLE MOVE NOT MVNL @MVNL 499	MVNL(499) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Trasferisce il complemento di due canali di dati nei canali specificati. S S+1 Stato del bit invertito D D+1	Uscita Richiesta
MOVE BIT MOVB @MOVB 082	MOVB(082) S C D **S:** dati o canale sorgente **C:** canale di controllo **D:** Canale di destinazione	Trasferisce il bit specificato. C m n S n D m	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
MOVE DIGIT MOVD @MOVD 083	MOVD(083) S C D **S:** dati o canale sorgente **C:** canale di controllo **D:** Canale di destinazione	Trasferisce la cifra o le cifre specificate (ogni cifra è composta da 4 bit).	Uscita Richiesta
MULTIPLE BIT TRANSFER XFRB @XFRB 062	XFRB(062) C S D **C:** canale di controllo **S:** primo canale sorgente **D:** primo canale di destinazione	Trasferisce il numero di bit consecutivi specificato.	Uscita Richiesta
BLOCK TRANSFER XFER @XFER 070	XFER(070) N S D **N:** Numero di canali **S:** primo canale sorgente **D:** primo canale di destinazione	Trasferisce il numero di canali consecutivi specificato.	Uscita Richiesta
BLOCK SET BSET @BSET 071	BSET(071) S St E **S:** canale sorgente **St:** canale di inizio **E:** canale di fine	Copia lo stesso canale in un intervallo di canali consecutivi.	Uscita Richiesta
DATA EXCHANGE XCHG @XCHG 073	XCHG(073) E1 E2 **E1:** primo canale di scambio **E2:** secondo canale di scambio	Scambia il contenuto dei due canali specificati.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE DATA EXCHANGE XCGL @XCGL 562	XCGL(562) E1 E2 **E1:** primo canale di scambio **E2:** secondo canale di scambio	Scambia il contenuto di una coppia di canali consecutivi con il contenuto di un'altra coppia di canali consecutivi. E1 E1+1 ⇄ E2 E2+1	Uscita Richiesta
SINGLE WORD DISTRIBUTE DIST @DIST 080	DIST(080) S Bs Of **S:** canale sorgente **Bs:** indirizzo base di destinazione **Of:** Offset	Trasferisce il canale sorgente in un canale di destinazione calcolato aggiungendo il valore di offset all'indirizzo di base. S, Bs, Of n, n, Bs+n	Uscita Richiesta
DATA COLLECT COLL @COLL 081	COLL(081) Bs Of D **Bs:** indirizzo base sorgente **Of:** Offset **D:** Canale di destinazione	Trasferisce il canale sorgente, calcolato aggiungendo un valore di offset all'indirizzo di base, nel canale di destinazione. Bs, Of n, n, Bs+n, D	Uscita Richiesta
MOVE TO REGISTER MOVR @MOVR 560	MOVR(560) S D **S:** sorgente (canale o bit desiderato) **D:** destinazione (registro indice)	Imposta l'indirizzo di memoria I/O interna del canale, del bit o del flag di completamento del temporizzatore/contatore specificato nel registro indice specificato. Usare l'istruzione MOVRW(561) per impostare l'indirizzo di memoria I/O interna relativo a un valore attuale del temporizzatore o del contatore in un registro indice. Indirizzo della memoria I/O di S S Registro indice D	Uscita Richiesta
MOVE TIMER/COUNTER PV TO REGISTER MOVRW @MOVRW 561	MOVRW(561) S D **S:** sorgente (numero TC desiderato) **D:** destinazione (registro indice)	Imposta l'indirizzo di memoria I/O interna del valore attuale del temporizzatore/contatore specificato nel registro indice specificato. Usare l'istruzione MOVR(560) per impostare l'indirizzo di memoria I/O interna relativo a un canale, a un bit o a un flag di completamento del temporizzatore/contatore in un registro indice. Indirizzo della memoria I/O di S S Solo valore attuale del temporizzatore/contatore Registro indice D	Uscita Richiesta

3-7 Istruzioni di scorrimento dei dati

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SHIFT REGISTER SFT 010	Ingresso dati / Ingresso di scorrimento / Ingresso di reset: SFT(010) / St / E **St:** canale di inizio **E:** canale di fine	Utilizza un registro di scorrimento. E / St+1, St+2 / St Dati persi Stato dell'ingresso dei dati per ogni ingresso di scorrimento	Uscita Richiesta
REVERSIBLE SHIFT REGISTER SFTR @SFTR 084	SFTR(084) / C / St / E **C:** canale di controllo **St:** canale di inizio **E:** canale di fine	Crea un registro di scorrimento che fa scorrere i dati a destra o a sinistra. Ingresso dati Ingresso dati Direzione di scorrimento	Uscita Richiesta
ASYNCHRONOUS SHIFT REGISTER ASFT @ASFT 017	ASFT(017) / C / St / E **C:** canale di controllo **St:** canale di inizio **E:** canale di fine	Fa scorrere tutti i dati diversi da zero all'interno dell'intervallo di canali specificato verso St o verso E e inserisce il valore esadecimale 0000 nei canali così liberati. Direzione di scorrimento Scorrimento abilitato Cancella Scorrimento Scorrimento Dati uguali a zero Dati diversi da zero	Uscita Richiesta
WORD SHIFT WSFT @WSFT 016	WSFT(016) / S / St / E **S:** canale sorgente **St:** canale di inizio **E:** canale di fine	Fa scorrere i dati tra St ed E, un canale alla volta. Dati persi	Uscita Richiesta
ARITHMETIC SHIFT LEFT ASL @ASL 025	ASL(025) / Wd **Wd:** Canale	Fa scorrere il contenuto del canale Wd di un bit a sinistra. CY	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE SHIFT LEFT ASLL @ASLL 570	ASLL(570) Wd **Wd:** Canale	Fa scorrere il contenuto dei canali Wd e Wd+1 di un bit a sinistra.	Uscita Richiesta
ARITHMETIC SHIFT RIGHT ASR @ASR 026	ASR(026) Wd **Wd:** Canale	Fa scorrere il contenuto del canale Wd di un bit a destra.	Uscita Richiesta
DOUBLE SHIFT RIGHT ASRL @ASRL 571	ASRL(571) Wd **Wd:** Canale	Fa scorrere il contenuto dei canali Wd e Wd+1 di un bit a destra.	Uscita Richiesta
ROTATE LEFT ROL @ROL 027	ROL(027) Wd **Wd:** Canale	Fa scorrere tutti i bit del canale Wd di un bit a sinistra, incluso il flag di riporto (CY).	Uscita Richiesta
DOUBLE ROTATE LEFT ROLL @ROLL 572	ROLL(572) Wd **Wd:** Canale	Fa scorrere tutti i bit dei canali Wd e Wd +1 di un bit a sinistra, incluso il flag di riporto (CY).	Uscita Richiesta
ROTATE LEFT WITHOUT CARRY RLNC @RLNC 574	RLNC(574) Wd **Wd:** Canale	Fa scorrere tutti i bit del canale Wd di un bit a sinistra, escluso il flag di riporto (CY).	Uscita Richiesta
DOUBLE ROTATE LEFT WITHOUT CARRY RLNL @RLNL 576	RLNL(576) Wd **Wd:** Canale	Fa scorrere tutti i bit dei canali Wd e Wd +1 di un bit a sinistra, escluso il flag di riporto (CY).	Uscita Richiesta
ROTATE RIGHT ROR @ROR 028	ROR(028) Wd **Wd:** Canale	Fa scorrere tutti i bit del canale Wd di un bit a destra, incluso il flag di riporto (CY)	Uscita Richiesta
DOUBLE ROTATE RIGHT RORL @RORL 573	RORL(573) Wd **Wd:** Canale	Fa scorrere tutti i bit dei canali Wd e Wd +1 di un bit a destra, incluso il flag di riporto (CY).	Uscita Richiesta

Istruzione / Codice mnemonico / Codice	Simbolo/ operando	Funzione	Posizione / Condizione di esecuzione
ROTATE RIGHT WITHOUT CARRY RRNC @RRNC 575	RRNC(575) Wd **Wd:** Canale	Fa scorrere tutti i bit del canale Wd di un bit a destra, escluso il flag di riporto (CY). Il contenuto del bit più a destra del canale Wd si sposta nel bit più a sinistra e nel flag di riporto (CY).	Uscita Richiesta
DOUBLE ROTATE RIGHT WITHOUT CARRY RRNL @RRNL 577	RRNL(577) Wd **Wd:** Canale	Fa scorrere tutti i bit dei canali Wd e Wd +1 di un bit a destra, escluso il flag di riporto (CY). Il contenuto del bit più a destra del canale Wd+1 si sposta nel bit più a sinistra del canale Wd e nel flag di riporto (CY).	Uscita Richiesta
ONE DIGIT SHIFT LEFT SLD @SLD 074	SLD(074) St E **St:** canale di inizio **E:** canale di fine	Fa scorrere i dati di una cifra (4 bit) a sinistra.	Uscita Richiesta
ONE DIGIT SHIFT RIGHT SRD @SRD 075	SRD(075) St E **St:** canale di inizio **E:** canale di fine	Fa scorrere i dati di una cifra (4 bit) a destra.	Uscita Richiesta
SHIFT N-BIT DATA LEFT NSFL @NSFL 578	NSFL(578) D C N **D:** canale iniziale per lo scorrimento **C:** bit iniziale **N:** lunghezza dei dati di scorrimento	Fa scorrere a sinistra il numero di bit specificato.	Uscita Richiesta
SHIFT N-BIT DATA RIGHT NSFR @NSFR 579	NSFR(579) D C N **D:** canale iniziale per lo scorrimento **C:** bit iniziale **N:** lunghezza dei dati di scorrimento	Fa scorrere a destra il numero di bit specificato.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SHIFT N-BITS LEFT NASL @NASL 580	NASL(580) D C **D:** canale di scorrimento **C:** canale di controllo	Fa scorrere i 16 bit specificati dei dati canale a sinistra del numero di bit specificato. 15 12 11 8 7 4 3 0 C 0 Scorrimento di n bit D a Dati persi CY D a Contenuto di "a" o "0" spostato a sinistra N bit	Uscita Richiesta
DOUBLE SHIFT N-BITS LEFT NSLL @NSLL 582	NSLL(582) D C **D:** canale di scorrimento **C:** canale di controllo	Fa scorrere i 32 bit specificati dei dati canale a sinistra del numero di bit specificato. 15 12 11 8 7 4 3 0 C 0 Scorrimento di n bit D+1 D a Contenuto di "a" o "0" spostato a sinistra Dati persi CY a N bit	Uscita Richiesta
SHIFT N-BITS RIGHT NASR @NASR 581	NASR(581) D C **D:** canale di scorrimento **C:** canale di controllo	Fa scorrere i 16 bit specificati dei dati canale a destra del numero di bit specificato. a D Contenuto di "a" o "0" spostato a sinistra a D CY Dati persi N bit	Uscita Richiesta
DOUBLE SHIFT N-BITS RIGHT NSRL @NSRL 583	NSRL(583) D C **D:** canale di scorrimento **C:** canale di controllo	Fa scorrere i 32 bit specificati dei dati canale a destra del numero di bit specificato. 15 12 11 8 7 4 3 0 C 0 Scorrimento di n bit a D+1 D Contenuto di "a" o "0" spostato a sinistra a CY Dati persi	Uscita Richiesta

3-8 Istruzioni di incremento e decremento

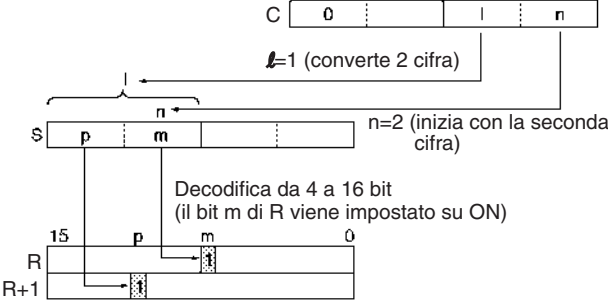
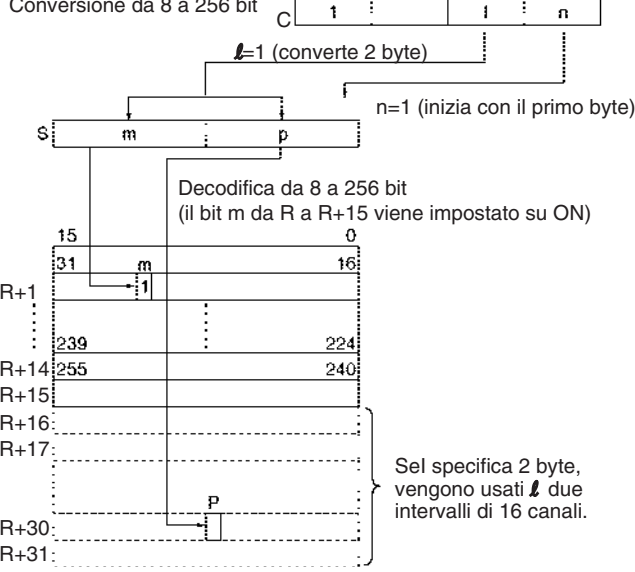
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
INCREMENT BINARY ++ @++ 590	++(590) Wd **Wd:** Canale	Incrementa di 1 il contenuto esadecimale a 4 cifre del canale specificato. Wd +1 → Wd	Uscita Richiesta
DOUBLE INCREMENT BINARY ++L @++L 591	++L(591) Wd **Wd:** Canale	Incrementa di 1 il contenuto esadecimale a 8 cifre dei canali specificati. Wd+1 Wd +1 → Wd+1 Wd	Uscita Richiesta
DECREMENT BINARY – – @– – 592	– – (592) Wd **Wd:** Canale	Decrementa di 1 il contenuto esadecimale a 4 cifre del canale specificato. Wd –1 → Wd	Uscita Richiesta
DOUBLE DECREMENT BINARY – –L @– –L 593	– – L(593) Wd **Wd:** primo canale	Decrementa di 1 il contenuto esadecimale a 8 cifre dei canali specificati. Wd+1 Wd –1 → Wd+1 Wd	Uscita Richiesta
INCREMENT BCD ++B @++B 594	++B(594) Wd **Wd:** Canale	Incrementa di 1 il contenuto in formato BCD a 4 cifre del canale specificato. Wd +1 → Wd	Uscita Richiesta
DOUBLE INCREMENT BCD ++BL @++BL 595	++BL(595) Wd **Wd:** primo canale	Incrementa di 1 il contenuto in formato BCD a 8 cifre dei canali specificati. Wd+1 Wd +1 → Wd+1 Wd	Uscita Richiesta
DECREMENT BCD – –B @– –B 596	– – B(596) Wd **Wd:** Canale	Decrementa di 1 il contenuto in formato BCD a 4 cifre del canale specificato. Wd –1 → Wd	Uscita Richiesta
DOUBLE DECREMENT BCD – –BL @– –BL 597	– – BL(597) Wd **Wd:** primo canale	Decrementa di 1 il contenuto in formato BCD a 8 cifre dei canali specificati. Wd+1 Wd –1 → Wd+1 Wd	Uscita Richiesta

3-9 Istruzioni matematiche con simboli

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SIGNED BINARY ADD WITHOUT CARRY + @+ 400	+(400) Au Ad R **Au:** canale addendo **Ad:** canali addendo **R:** canale del risultato	Somma dati e/o costanti esadecimali a 4 cifre (canale singolo). Au — Binario con segno + Ad — Binario con segno Il flag CY viene attivato in presenza di riporto. CY R — Binario con segno	Uscita Richiesta
DOUBLE SIGNED BINARY ADD WITHOUT CARRY +L @+L 401	+L(401) Au Ad R **Au:** primo canale addendo **Ad:** primo canale addendo **R:** primo canale del risultato	Somma dati e/o costanti esadecimali a 8 cifre (doppio canale). Au+1 Au — Binario con segno + Ad+1 Ad — Binario con segno Il flag CY viene attivato in presenza di riporto. CY R+1 R — Binario con segno	Uscita Richiesta
SIGNED BINARY ADD WITH CARRY +C @+C 402	+C(402) Au Ad R **Au:** canale addendo **Ad:** canali addendo **R:** canale del risultato	Somma dati e/o costanti esadecimali a 4 cifre (canale singolo), incluso il flag di riporto (CY). Au — Binario con segno Ad — Binario con segno + CY Il flag CY viene attivato in presenza di riporto. CY R — Binario con segno	Uscita Richiesta
DOUBLE SIGNED BINARY ADD WITH CARRY +CL @+CL 403	+CL(403) Au Ad R **Au:** primo canale addendo **Ad:** primo canale addendo **R:** primo canale del risultato	Somma dati e/o costanti esadecimali a 8 cifre (doppio canale), incluso il flag di riporto (CY). Au+1 Au — Binario con segno Ad+1 Ad — Binario con segno + CY Il flag CY viene attivato in presenza di riporto. CY R+1 R — Binario con segno	Uscita Richiesta
BCD ADD WITHOUT CARRY +B @+B 404	+B(404) Au Ad R **Au:** canale addendo **Ad:** canali addendo **R:** canale del risultato	Somma dati e/o costanti in formato BCD a 4 cifre (canale singolo). Au (BCD) + Ad (BCD) Il flag CY viene attivato in presenza di riporto. CY R (BCD)	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE BCD ADD WITHOUT CARRY +BL @+BL 405	+BL(405) Au Ad R **Au:** primo canale addendo **Ad:** primo canale addendo **R:** primo canale del risultato	Somma dati e/o costanti in formato BCD a 8 cifre (doppio canale). Au+1 Au (BCD) + Ad+1 Ad (BCD) Il flag CY viene attivato in presenza di riporto. CY R+1 R (BCD)	Uscita Richiesta
BCD ADD WITH CARRY +BC @+BC 406	+BC(406) Au Ad R **Au:** canale addendo **Ad:** canali addendo **R:** canale del risultato	Somma dati e/o costanti in formato BCD a 4 cifre (canale singolo), incluso il flag di riporto (CY). Au (BCD) Ad (BCD) + CY Il flag CY viene attivato in presenza di riporto. CY R (BCD)	Uscita Richiesta
DOUBLE BCD ADD WITH CARRY +BCL @+BCL 407	+BCL(407) Au Ad R **Au:** primo canale addendo **Ad:** primo canale addendo **R:** primo canale del risultato	Somma dati e/o costanti in formato BCD a 8 cifre (doppio canale), incluso il flag di riporto (CY). Au+1 Au (BCD) Ad+1 Ad (BCD) + CY Il flag CY viene attivato in presenza di riporto. CY R+1 R (BCD)	Uscita Richiesta
SIGNED BINARY SUBTRACT WITHOUT CARRY – @– 410	–(410) Mi Su R **Mi:** canale minuendo **Su:** canale sottraendo **R:** canale del risultato	Esegue una sottrazione tra dati e/o costanti esadecimali a 4 cifre (canale singolo). Mi Binario con segno – Su Binario con segno Il flag CY viene attivato in presenza di prestito. CY R Binario con segno	Uscita Richiesta
DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY –L @–L 411	–L(411) Mi Su R **Mi:** canale minuendo **Su:** canale sottraendo **R:** canale del risultato	Esegue una sottrazione tra dati e/o costanti esadecimali a 8 cifre (doppio canale). Mi+1 Mi Binario con segno – Su+1 Su Binario con segno Il flag CY viene attivato in presenza di prestito. CY R+1 R Binario con segno	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SIGNED BINARY SUBTRACT WITH CARRY −C @−C 412	−C(412) Mi Su R **Mi:** canale minuendo **Su:** canale sottraendo **R:** canale del risultato	Esegue una sottrazione tra dati e/o costanti esadecimali a 4 cifre (canale singolo), incluso il flag di riporto (CY). Mi — Binario con segno Su — Binario con segno – CY CY R — Binario con segno Il flag CY viene attivato in presenza di prestito.	Uscita Richiesta
DOUBLE SIGNED BINARY WITH CARRY −CL @−CL 413	−CL(413) Mi Su R **Mi:** canale minuendo **Su:** canale sottraendo **R:** canale del risultato	Esegue una sottrazione tra dati e/o costanti esadecimali a 8 cifre (doppio canale), incluso il flag di riporto (CY). Mi+1 Mi — Binario con segno Su+1 Su — Binario con segno – CY CY R+1 R — Binario con segno Il flag CY viene attivato in presenza di prestito.	Uscita Richiesta
BCD SUBTRACT WITHOUT CARRY −B @−B 414	−B(414) Mi Su R **Mi:** canale minuendo **Su:** canale sottraendo **R:** canale del risultato	Esegue una sottrazione tra dati e/o costanti in formato BCD a 4 cifre (canale singolo). Mi (BCD) – Su (BCD) CY R (BCD) Il flag CY viene attivato in presenza di riporto.	Uscita Richiesta
DOUBLE BCD SUBTRACT WITHOUT CARRY −BL @−BL 415	−BL(415) Mi Su R **Mi:** primo canale minuendo **Su:** primo canale sottraendo **R:** primo canale del risultato	Esegue una sottrazione tra dati e/o costanti in formato BCD a 8 cifre (doppio canale). Mi +1 Mi (BCD) – Su+1 Su (BCD) CY R+1 R (BCD) Il flag CY viene attivato in presenza di prestito.	Uscita Richiesta
BCD SUBTRACT WITH CARRY −BC @−BC 416	−BC(416) Mi Su R **Mi:** canale minuendo **Su:** canale sottraendo **R:** canale del risultato	Esegue una sottrazione tra dati e/o costanti in formato BCD a 4 cifre (canale singolo), incluso il flag di riporto (CY). Mi (BCD) Su (BCD) – CY CY R (BCD) Il flag CY viene attivato in presenza di prestito.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE BCD SUBTRACT WITH CARRY −BCL @−BCL 417	−BCL(417) Mi Su R **Mi:** primo canale minuendo **Su:** primo canale sottraendo **R:** primo canale del risultato	Esegue una sottrazione tra dati e/o costanti in formato BCD a 8 cifre (doppio canale), incluso il flag di riporto (CY). Mi +1 | Mi (BCD) Su+1 | Su (BCD) – CY Il flag CY viene attivato in presenza di prestito. CY | R+1 | R (BCD)	Uscita Richiesta
SIGNED BINARY MULTIPLY ∗ @∗ 420	*(420) Md Mr R **Md:** canale moltiplicando **Mr:** canale moltiplicatore **R:** canale del risultato	Moltiplica dati e/o costanti esadecimali a 4 cifre con segno. Md Binario con segno × Mr Binario con segno R +1 | R Binario con segno	Uscita Richiesta
DOUBLE SIGNED BINARY MULTIPLY ∗L @∗L 421	*L(421) Md Mr R **Md:** primo canale moltiplicando **Mr:** primo canale moltiplicatore **R:** primo canale del risultato	Moltiplica dati e/o costanti esadecimali a 8 cifre con segno. Md + 1 | Md Binario con segno × Mr + 1 | Mr Binario con segno R +3 | R + 2 | R +1 | R Binario con segno	Uscita Richiesta
UNSIGNED BINARY MULTIPLY ∗U @∗U 422	*U(422) Md Mr R **Md:** canale moltiplicando **Mr:** canale moltiplicatore **R:** canale del risultato	Moltiplica dati e/o costanti esadecimali a 4 cifre senza segno. Md Binario senza segno × Mr Binario senza segno R +1 | R Binario senza segno	Uscita Richiesta
DOUBLE UNSIGNED BINARY MULTIPLY ∗UL @∗UL 423	*UL(423) Md Mr R **Md:** primo canale moltiplicando **Mr:** primo canale moltiplicatore **R:** primo canale del risultato	Moltiplica dati e/o costanti esadecimali a 8 cifre senza segno. Md + 1 | Md Binario senza segno × Mr + 1 | Mr Binario senza segno R +3 | R + 2 | R +1 | R Binario senza segno	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
BCD MULTIPLY ∗B @∗B 424	*B(424) Md Mr R **Md:** canale moltiplicando **Mr:** canale moltiplicatore **R:** canale del risultato	Moltiplica dati e/o costanti in formato BCD a 4 cifre (canale singolo). Md (BCD) × Mr (BCD) R +1, R (BCD)	Uscita Richiesta
DOUBLE BCD MULTIPLY ∗BL @∗BL 425	*BL(425) Md Mr R **Md:** primo canale moltiplicando **Mr:** primo canale moltiplicatore **R:** primo canale del risultato	Moltiplica dati e/o costanti in formato BCD a 8 cifre (doppio canale). Md + 1, Md (BCD) × Mr + 1, Mr (BCD) R +3, R + 2, R +1, R (BCD)	Uscita Richiesta
SIGNED BINARY DIVIDE / @/ 430	/(430) Dd Dr R **Dd:** canale dividendo **Dr:** canale divisore **R:** canale del risultato	Esegue una divisione tra dati e/o costanti esadecimali a 4 cifre (canale singolo) con segno. Dd Binario con segno ÷ Dr Binario con segno R +1, R Binario con segno Resto Quoziente	Uscita Richiesta
DOUBLE SIGNED BINARY DIVIDE /L @/L 431	/L(431) Dd Dr R **Dd:** primo canale dividendo **Dr:** primo canale divisore **R:** primo canale del risultato	Esegue una divisione tra dati e/o costanti esadecimali a 8 cifre (doppio canale) con segno. Dd + 1, Dd Binario con segno ÷ Dr + 1, Dr Binario con segno R +3, R + 2, R +1, R Binario con segno Resto Quoziente	Uscita Richiesta
UNSIGNED BINARY DIVIDE /U @/U 432	/U(432) Dd Dr R **Dd:** canale dividendo **Dr:** canale divisore **R:** canale del risultato	Esegue una divisione tra dati e/o costanti esadecimali a 4 cifre (canale singolo) senza segno. Dd Binario senza segno ÷ Dr Binario senza segno R +1, R Binario senza segno Resto Quoziente	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE UNSIGNED BINARY DIVIDE /UL @/UL 433	/UL(433) Dd Dr R **Dd:** primo canale dividendo **Dr:** primo canale divisore **R:** primo canale del risultato	Esegue una divisione tra dati e/o costanti esadecimali a 8 cifre (doppio canale) senza segno. Dd + 1 | Dd — Binario senza segno ÷ Dr + 1 | Dr — Binario senza segno R +3 | R + 2 | R +1 | R — Binario senza segno Resto — Quoziente	Uscita Richiesta
BCD DIVIDE /B @/B 434	/B(434) Dd Dr R **Dd:** canale dividendo **Dr:** canale divisore **R:** canale del risultato	Esegue una divisione tra dati e/o costanti in formato BCD a 4 cifre (canale singolo). Dd (BCD) ÷ Dr (BCD) R +1 | R (BCD) Resto — Quoziente	Uscita Richiesta
DOUBLE BCD DIVIDE /BL @/BL 435	/BL(435) Dd Dr R **Dd:** primo canale dividendo **Dr:** primo canale divisore **R:** primo canale del risultato	Esegue una divisione tra dati e/o costanti in formato BCD a 8 cifre (doppio canale). Dd + 1 | Dd (BCD) ÷ Dr + 1 | Dr (BCD) R +3 | R + 2 | R +1 | R (BCD) Resto — Quoziente	Uscita Richiesta

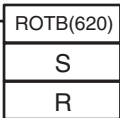
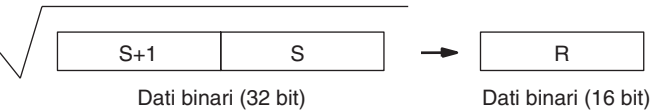
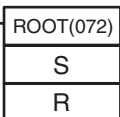
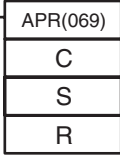
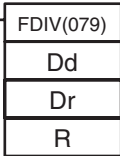
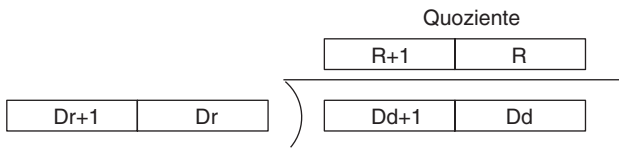
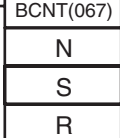
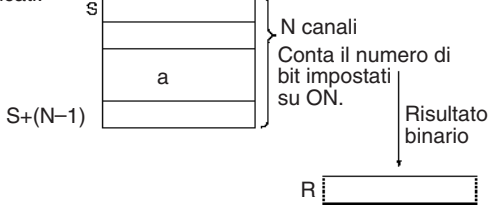
3-10 Istruzioni di conversione

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
BCD-TO-BINARY BIN @BIN 023	BIN(023) S R **S:** canale sorgente **R:** canale del risultato	Converte dati in formato BCD in dati binari. S (BCD) → R Binario	Uscita Richiesta
DOUBLE BCD-TO-DOUBLE BINARY BINL @BINL 058	BINL(058) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte dati in formato BCD a 8 cifre in dati esadecimali a 8 cifre (binari a 32 bit). S (BCD) → R Binario S+1 (BCD) → R+1 Binario	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
BINARY-TO-BCD BCD @BCD 024	BCD(024) S R **S:** canale sorgente **R:** canale del risultato	Converte un canale di dati binari in un canale di dati in formato BCD. S Binario → R (BCD)	Uscita Richiesta
DOUBLE BINARY-TO-DOUBLE BCD BCDL @BCDL 059	BCDL(059) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte dati esadecimali a 8 cifre (binari a 32 bit) in dati in formato BCD a 8 cifre. S Binario, S+1 Binario → R (BCD), R+1 (BCD)	Uscita Richiesta
2'S COMPLEMENT NEG @NEG 160	NEG(160) S R **S:** canale sorgente **R:** canale del risultato	Calcola il complemento a 2 di un canale di dati esadecimali. Complemento a 2 (complemento + 1) $\overline{(S)}$ → (R)	Uscita Richiesta
DOUBLE 2'S COMPLEMENT NEGL @NEGL 161	NEGL(161) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il complemento a 2 di due canali di dati esadecimali. Complemento a 2 (complemento + 1) $\overline{(S+1, S)}$ → (R+1, R)	Uscita Richiesta
16-BIT TO 32-BIT SIGNED BINARY SIGN @SIGN 600	SIGN(600) S R **S:** canale sorgente **R:** primo canale del risultato	Converte un valore binario con segno a 16 bit nell'equivalente a 32 bit. MSB S MSB = 1: FFFF esadecimale MSB = 0: 0000 esadecimale D+1 D D = contenuto di S	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DATA DECODER MLPX @MLPX 076	MLPX(076) S C R **S:** canale sorgente **C:** canale di controllo **R:** primo canale del risultato	Legge il valore numerico nella cifra o nel byte specificato e lo copia nel canale sorgente, imposta su ON il bit corrispondente e su OFF tutti gli altri bit nel canale del risultato o nell'intervallo di 16 canali. Conversione da 4 a 16 bit Conversione da 8 a 256 bit	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DATA ENCODER DMPX @DMPX 077	DMPX(077) S R C **S:** primo canale sorgente **R:** canale del risultato **C:** canale di controllo	Determina la posizione del primo o dell'ultimo bit impostato su ON all'interno del canale sorgente o dell'intervallo di 16 canali e scrive tale valore nella cifra o nel byte specificato nel canale del risultato.	Uscita Richiesta
ASCII CONVERT ASC @ASC 086	ASC(086) S Di D **S:** canale sorgente **Di:** identificatore della cifra **D:** primo canale di destinazione	Converte cifre esadecimali a 4 bit nel canale sorgente negli equivalenti ASCII a 8 bit.	Uscita Richiesta

Istruzione / Codice mnemonico / Codice	Simbolo/ operando	Funzione	Posizione / Condizione di esecuzione
ASCII TO HEX HEX @HEX 162	HEX(162) S Di D **S:** primo canale sorgente **Di:** identificatore della cifra **D:** Canale di destinazione	Converte fino a 4 byte di dati ASCII all'interno del canale sorgente negli equivalenti esadecimali e scrive tali cifre nel canale di destinazione specificato.	Uscita Richiesta
COLUMN TO LINE LINE @LINE 063	LINE(063) S N D **S:** primo canale sorgente **N:** numero bit **D:** Canale di destinazione	Converte una colonna di bit all'interno di un intervallo di 16 canali (lo stesso numero di bit in 16 canali consecutivi) nei 16 bit del canale di destinazione	Uscita Richiesta
LINE TO COLUMN COLM @COLM 064	COLM(064) S D N **S:** canale sorgente **D:** primo canale di destinazione **N:** numero bit	Converte i 16 bit del canale sorgente in una colonna di bit all'interno di un intervallo di 16 canali di destinazione (lo stesso numero di bit in 16 canali consecutivi).	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SIGNED BCD-TO-BINARY BINS @BINS 470	BINS(470) C S D **C:** canale di controllo **S:** canale sorgente **D:** Canale di destinazione	Converte un canale di dati in formato BCD con segno in un canale di dati binari con segno. C Formato BCD con segno specificato in C S BCD con segno → D Binario con segno	Uscita Richiesta
DOUBLE SIGNED BCD-TO-BINARY BISL @BISL 472	BISL(472) C S D **C:** canale di controllo **S:** primo canale sorgente **D:** primo canale di destinazione	Converte dati in formato BCD con segno in doppia precisione in dati binari con segno in doppia precisione. C Formato BCD con segno specificato in C S BCD con segno → D Binario con segno S+1 BCD con segno → D+1 Binario con segno	Uscita Richiesta
SIGNED BINARY-TO-BCD BCDS @BCDS 471	BCDS(471) C S D **C:** canale di controllo **S:** canale sorgente **D:** Canale di destinazione	Converte un canale di dati binari con segno in un canale di dati in formato BCD con segno. C Formato BCD con segno specificato in C S Binario con segno → D BCD con segno	Uscita Richiesta
DOUBLE SIGNED BINARY-TO-BCD BDSL @BDSL 473	BDSL(473) C S D **C:** canale di controllo **S:** primo canale sorgente **D:** primo canale di destinazione	Converte dati binari con segno in doppia precisione in dati in formato BCD con segno in doppia precisione. C Formato BCD con segno specificato in C S Binario con segno → D BCD con segno S+1 Binario con segno → D+1 BCD con segno	Uscita Richiesta
GRAY CODE CONVERSION GRY 474 (solo CPU della serie CS/CJ versione 2.0 o successiva, comprese le CPU CS1-H, CJ1-H e CJ1M con numero di lotto 030201 e successivo)	GRY(474) C S D **C:** canale di controllo **S:** canale sorgente **D:** primo canale di destinazione	Converte i dati in codice Gray nel canale specificato in dati in formato binario, BCD o sotto forma di angolo (°) alla risoluzione specificata.	Uscita Richiesta

3-11 Istruzioni logiche

<table>
<tr><th>Istruzione
Codice mnemonico
Codice</th><th>Simbolo/ operando</th><th>Funzione</th><th>Posizione
Condizione di esecuzione</th></tr>
<tr><td>LOGICAL AND
ANDW
@ANDW
034</td><td>ANDW(034)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un AND logico tra i bit corrispondenti nei canali singoli di dati canale e/o costanti.
$I_1 . I_2 \rightarrow R$
<table>
<tr><th>I_1</th><th>I_2</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>Uscita
Richiesta</td></tr>
<tr><td>DOUBLE LOGICAL AND
ANDL
@ANDL
610</td><td>ANDL(610)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un AND logico tra i bit corrispondenti nei canali doppi di dati canale e/o costanti.
$(I_1, I_1+1) . (I_2, I_2+1) \rightarrow (R, R+1)$
<table>
<tr><th>I_1, I_1+1</th><th>I_2, I_2+1</th><th>R, R+1</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>0</td></tr>
<tr><td>0</td><td>1</td><td>0</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>Uscita
Richiesta</td></tr>
<tr><td>LOGICAL OR
ORW
@ORW
035</td><td>ORW(035)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un OR logico tra i bit corrispondenti nei canali singoli di dati canale e/o costanti.
$I_1 + I_2 \rightarrow R$
<table>
<tr><th>I_1</th><th>I_2</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>Uscita
Richiesta</td></tr>
<tr><td>DOUBLE LOGICAL OR
ORWL
@ORWL
611</td><td>ORWL(611)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un OR logico tra i bit corrispondenti nei canali doppi di dati canale e/o costanti.
$(I_1, I_1+1) + (I_2, I_2+1) \rightarrow (R, R+1)$
<table>
<tr><th>I_1, I_1+1</th><th>I_2, I_2+1</th><th>R, R+1</th></tr>
<tr><td>1</td><td>1</td><td>1</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>Uscita
Richiesta</td></tr>
<tr><td>EXCLUSIVE OR
XORW
@XORW
036</td><td>XORW(036)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un OR logico esclusivo tra i bit corrispondenti nei canali singoli di dati canale e/o costanti.
$I_1 . I_2 + I_1 . I_2 \rightarrow R$
<table>
<tr><th>I_1</th><th>I_2</th><th>R</th></tr>
<tr><td>1</td><td>1</td><td>0</td></tr>
<tr><td>1</td><td>0</td><td>1</td></tr>
<tr><td>0</td><td>1</td><td>1</td></tr>
<tr><td>0</td><td>0</td><td>0</td></tr>
</table></td><td>Uscita
Richiesta</td></tr>
</table>

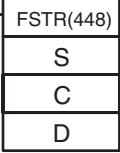
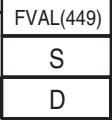
<table>
<tr><th>Istruzione
Codice mnemonico
Codice</th><th>Simbolo/ operando</th><th>Funzione</th><th>Posizione
Condizione di esecuzione</th></tr>
<tr><td>DOUBLE EXCLUSIVE OR
XORL
@XORL
612</td><td>XORL(612)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un OR logico esclusivo tra i bit corrispondenti nei canali doppi di dati canale e/o costanti.

$(I_1, I_1+1) . \overline{(I_2, I_2+1)} + \overline{(I_1, I_1+1)} . (I_2, I_2+1) \rightarrow (R, R+1)$
<table><tr><th>I_1, I_1+1</th><th>I_2, I_2+1</th><th>R, R+1</th></tr><tr><td>1</td><td>1</td><td>0</td></tr><tr><td>1</td><td>0</td><td>1</td></tr><tr><td>0</td><td>1</td><td>1</td></tr><tr><td>0</td><td>0</td><td>0</td></tr></table></td><td>Uscita
Richiesta</td></tr>
<tr><td>EXCLUSIVE NOR
XNRW
@XNRW
037</td><td>XNRW(037)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: canale del risultato</td><td>Esegue un NOR logico esclusivo tra i canali singoli corrispondenti di dati canale e/o costanti.

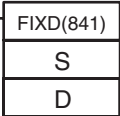
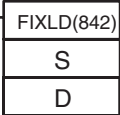
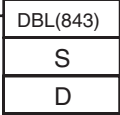
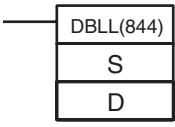
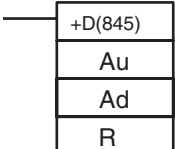
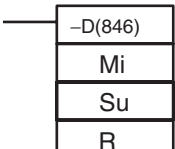
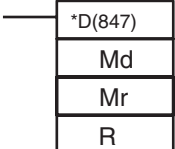
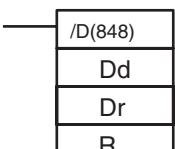
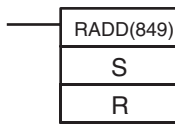
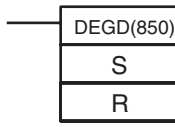
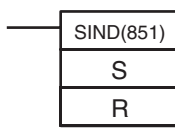
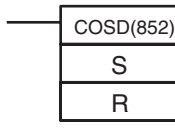
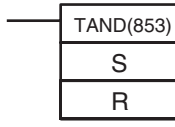
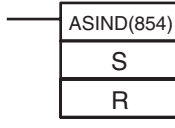
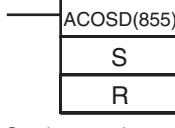
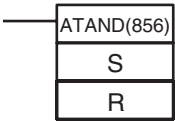
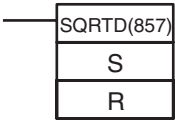
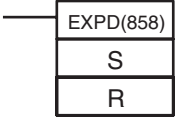
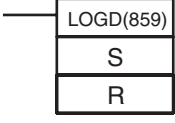
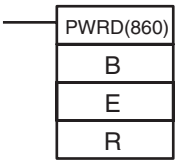
$I_1 . I_2 + \overline{I_1} . \overline{I_2} \rightarrow R$
<table><tr><th>I_1</th><th>I_2</th><th>R</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table></td><td>Uscita
Richiesta</td></tr>
<tr><td>DOUBLE EXCLUSIVE NOR
XNRL
@XNRL
613</td><td>XNRL(613)
I_1
I_2
R
I1: ingresso 1
I2: ingresso 2
R: primo canale del risultato</td><td>Esegue un NOR logico esclusivo tra i bit corrispondenti nei canali doppi di dati canale e/o costanti.

$(I_1, I_1+1) . (I_2, I_2+1) + \overline{(I_1, I_1+1)} . \overline{(I_2, I_2+1)} \rightarrow (R, R+1)$
<table><tr><th>I_1, I_1+1</th><th>I_2, I_2+1</th><th>R, R+1</th></tr><tr><td>1</td><td>1</td><td>1</td></tr><tr><td>1</td><td>0</td><td>0</td></tr><tr><td>0</td><td>1</td><td>0</td></tr><tr><td>0</td><td>0</td><td>1</td></tr></table></td><td>Uscita
Richiesta</td></tr>
<tr><td>COMPLEMENT
COM
@COM
029</td><td>COM(029)
Wd
Wd: Canale</td><td>Imposta su OFF tutti i bit il cui stato è ON e su ON tutti i bit il cui stato è OFF nel canale Wd.
$\overline{Wd} \rightarrow Wd$: 1 $\rightarrow$ 0 and 0 $\rightarrow$ 1</td><td>Uscita
Richiesta</td></tr>
<tr><td>DOUBLE COMPLEMENT
COML
@COML
614</td><td>COML(614)
Wd
Wd: Canale</td><td>Imposta su OFF tutti i bit il cui stato è ON e su ON tutti i bit il cui stato è OFF nei canali Wd e Wd+1.
$\overline{(Wd+1, Wd)} \rightarrow (Wd+1, Wd)$</td><td>Uscita
Richiesta</td></tr>
</table>

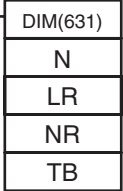
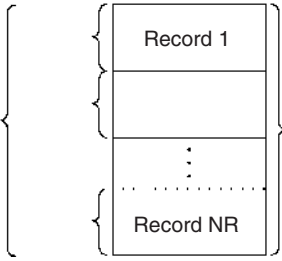
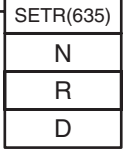
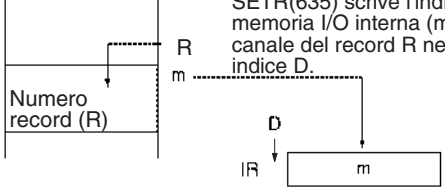
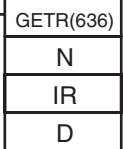
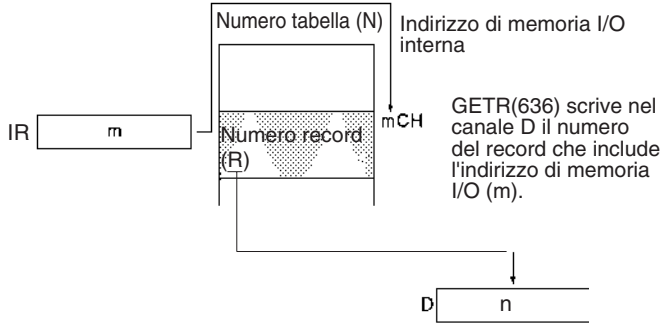
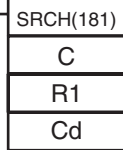
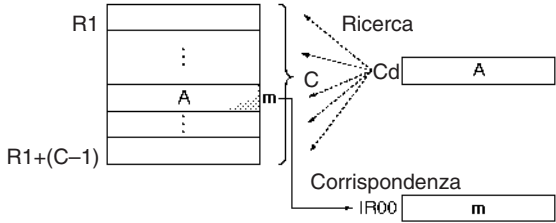
3-12 Istruzioni matematiche speciali

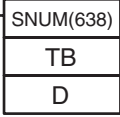
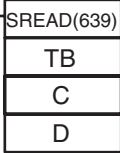
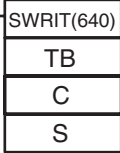
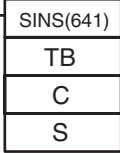
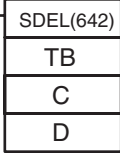
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
BINARY ROOT ROTB @ROTB 620	ROTB(620) S R **S:** primo canale sorgente **R:** canale del risultato	Calcola la radice quadrata del contenuto binario a 32 bit dei canali specificati e invia la parte intera del risultato al canale del risultato specificato. √ S+1 S → R Dati binari (32 bit) Dati binari (16 bit)	Uscita Richiesta
BCD SQUARE ROOT ROOT @ROOT 072	ROOT(072) S R **S:** primo canale sorgente **R:** canale del risultato	Calcola la radice quadrata di un numero in formato BCD a 8 cifre e invia la parte intera del risultato al canale del risultato specificato. √ S+1 S → R Dati in formato BCD (8 cifre) Dati in formato BCD (4 cifre)	Uscita Richiesta
ARITHMETIC PROCESS APR @APR 069	APR(069) C S R **C:** canale di controllo **S:** Dati sorgente **R:** canale del risultato	Calcola seno, coseno o un'estrapolazione lineare dei dati sorgente. La funzione di estrapolazione lineare consente di approssimare relazioni tra X e Y con segmenti di linea.	Uscita Richiesta
FLOATING POINT DIVIDE FDIV @FDIV 079	FDIV(079) Dd Dr R **Dd:** primo canale dividendo **Dr:** primo canale divisore **R:** primo canale del risultato	Divide un numero a virgola mobile a 7 cifre per un altro numero. I numeri a virgola mobile sono espressi in notazione scientifica (7 cifre di mantissa e 1 cifra di esponente). Quoziente R+1 R Dr+1 Dr) Dd+1 Dd	Uscita Richiesta
BIT COUNTER BCNT @BCNT 067	BCNT(067) N S R **N:** Numero di canali **S:** primo canale sorgente **R:** canale del risultato	Conta il numero totale di bit impostati su ON presenti nei canali specificati. S … a … S+(N–1) } N canali Conta il numero di bit impostati su ON. Risultato binario R	Uscita Richiesta

3-13 Istruzioni matematiche a virgola mobile

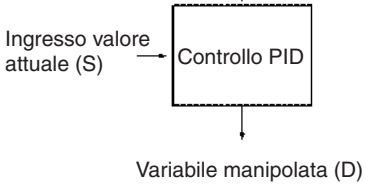
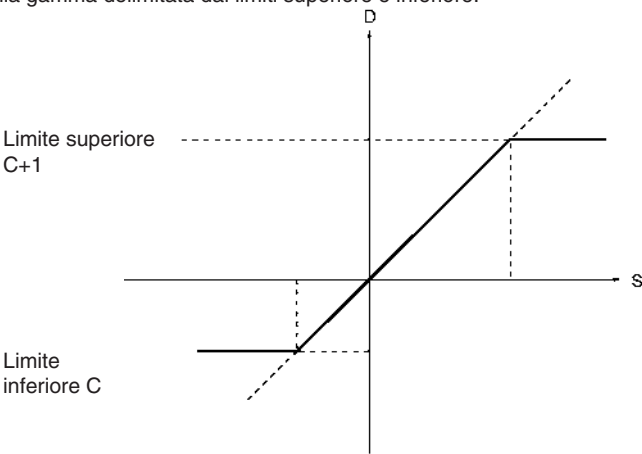
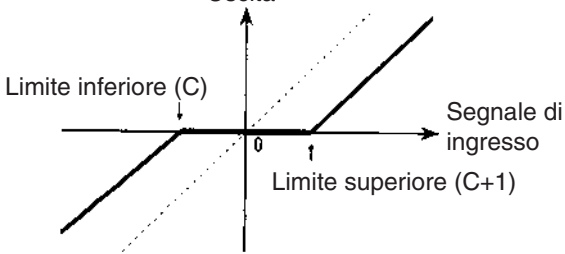
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
FLOATING TO 16-BIT FIX @FIX 450	FIX(450) S R **S:** primo canale sorgente **R:** canale del risultato	Converte un valore a virgola mobile a 32 bit in dati binari a 16 bit con segno e invia il risultato al canale del risultato specificato. S+1 | S — Dati a virgola mobile (32 bit) ↓ R — Dati binari con segno (16 bit)	Uscita Richiesta
FLOATING TO 32-BIT FIXL @FIXL 451	FIXL(451) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte un valore a virgola mobile a 32 bit in dati binari a 32 bit con segno e invia il risultato ai canali del risultato specificati. S+1 | S — Dati a virgola mobile (32 bit) ↓ R+1 | R — Dati binari con segno (32 bit)	Uscita Richiesta
16-BIT TO FLOATING FLT @FLT 452	FLT(452) S R **S:** canale sorgente **R:** primo canale del risultato	Converte un valore binario a 16 bit con segno in dati a virgola mobile a 32 bit e invia il risultato ai canali del risultato specificati. S — Dati binari con segno (16 bit) ↓ R+1 | R — Dati a virgola mobile (32 bit)	Uscita Richiesta
32-BIT TO FLOATING FLTL @FLTL 453	FLTL(453) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte un valore binario a 32 bit con segno in dati a virgola mobile a 32 bit e invia il risultato ai canali del risultato specificati. S+1 | S — Dati binari con segno (32 bit) ↓ R+1 | R — Dati a virgola mobile (32 bit)	Uscita Richiesta
FLOATING-POINT ADD +F @+F 454	+F(454) Au Ad R **Au:** primo canale addendo **AD:** primo canale addendo **R:** primo canale del risultato	Somma due numeri a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. Au+1 | Au — Addendo (dati a virgola mobile a 32 bit) + Ad+1 | Ad — Addendo (dati a virgola mobile a 32 bit) R+1 | R — Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
FLOATING-POINT SUBTRACT –F @–F 455	F(455) Mi Su R **Mi:** primo canale minuendo **Su:** primo canale sottraendo **R:** primo canale del risultato	Esegue la sottrazione tra due numeri a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. Mi+1 | Mi — Minuendo (dati a virgola mobile a 32 bit) – Su+1 | Su — Sottraendo (dati a virgola mobile a 32 bit) R+1 | R — Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
FLOATING-POINT MULTIPLY ∗F @∗F 456	∗F(456) Md Mr R **Md:** primo canale moltiplicando **Mr:** primo canale moltiplicatore **R:** primo canale del risultato	Moltiplica due numeri a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. Md+1 | Md — Moltiplicando (dati a virgola mobile a 32 bit) × Mr+1 | Mr — Moltiplicatore (dati a virgola mobile a 32 bit) R+1 | R — Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
FLOATING-POINT DIVIDE /F @/F 457	∗F(457) Dd Dr R **Dd:** primo canale dividendo **Dr:** primo canale divisore **R:** primo canale del risultato	Esegue la divisione tra due numeri a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. Dd+1 | Dd — Dividendo (dati a virgola mobile a 32 bit) ÷ Dr+1 | Dr — Divisore (dati a virgola mobile a 32 bit) R+1 | R — Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
DEGREES TO RADIANS RAD @RAD 458	RAD(458) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte un numero a virgola mobile a 32 bit da gradi a radianti e invia il risultato nei canali del risultato specificati. S+1 | S — Sorgente (gradi, dati a virgola mobile a 32 bit) ↓ R+1 | R — Risultato (radianti, dati a virgola mobile a 32 bit)	Uscita Richiesta
RADIANS TO DEGREES DEG @DEG 459	DEG(459) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte un numero a virgola mobile a 32 bit da radianti a gradi e invia il risultato nei canali del risultato specificati. S+1 | S — Sorgente (radianti, dati a virgola mobile a 32 bit) ↓ R+1 | R — Risultato (gradi, dati a virgola mobile a 32 bit)	Uscita Richiesta
SINE SIN @SIN 460	SIN(460) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il seno di un numero a virgola mobile a 32 bit in radianti e invia il risultato nei canali del risultato specificati. SIN (S+1 | S) — Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R — Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
COSINE COS @COS 461	COS(461) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il coseno di un numero a virgola mobile a 32 bit in radianti e invia il risultato nei canali del risultato specificati. COS(S+1 | S) — Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R — Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
TANGENT TAN @TAN 462	TAN(462) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola la tangente di un numero a virgola mobile a 32 bit in radianti e invia il risultato nei canali del risultato specificati. TAN (S+1 | S) Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
ARC SINE ASIN @ASIN 463	ASIN(463) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola l'arcoseno di un numero a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. La funzione di arcoseno è l'inverso della funzione di seno e restituisce l'angolo che produce un dato valore di seno compreso tra -1 e 1. SIN^{-1} (S+1 | S) Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
ARC COSINE ACOS @ACOS 464	ACOS(464) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola l'arcocoseno di un numero a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. La funzione di arcocoseno è l'inverso della funzione di coseno e restituisce l'angolo che produce un dato valore di coseno compreso tra -1 e 1. COS^{-1} (S+1 | S) Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
ARC TANGENT ATAN @ATAN 465	ATAN(465) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola l'arcotangente di un numero a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. La funzione di arcotangente è l'inverso della funzione di tangente e restituisce l'angolo che produce un dato valore di tangente. TAN^{-1} (S+1 | S) Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
SQUARE ROOT SQRT @SQRT 466	SQRT(466) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola la radice quadrata di un numero a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. √ S+1 | S Sorgente (dati a virgola mobile a 32 bit) ↓ R+1 | R Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
EXPONENT EXP @EXP 467	EXP(467) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il valore esponenziale naturale (in base e) di un numero a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. e: S+1, S → Sorgente (dati a virgola mobile a 32 bit) R+1, R → Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
LOGARITHM LOG @LOG 468	LOG(468) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il logaritmo naturale (in base e) di un numero a virgola mobile a 32 bit e invia il risultato nei canali del risultato specificati. $\log_e$: S+1, S → Sorgente (dati a virgola mobile a 32 bit) R+1, R → Risultato (dati a virgola mobile a 32 bit)	Uscita Richiesta
EXPONENTIAL POWER PWR @PWR 840	PWR(840) B E R **B:** primo canale base **E:** primo canale esponente **R:** primo canale del risultato	Eleva un numero a virgola mobile a 32 bit alla potenza di un altro numero a virgola mobile a 32 bit. E+1, E: Alimentazione B+1, S: Base → R+1, R	Uscita Richiesta
FLOATING SYMBOL COMPARISON (solo CS1-H, CJ1-H, CJ1M o CS1D) LD, AND o OR + =F (329), <>F (330), <F (331), <=F (332), >F (333) o >=F (334)	Uso di LD: Simbolo, opzione S1 S2 Uso di AND: Simbolo, opzione S1 S2 Uso di OR: Simbolo, opzione S1 S2 **S1:** dati di confronto 1 **S2:** dati di confronto 2	Confronta i dati in singola precisione (32 bit) oppure le constanti specificate e crea una condizione di esecuzione ON se il risultato del confronto è vero. Con le istruzioni di confronto con simboli a virgola mobile è possibile utilizzare tre tipi di simboli: LD (LOAD), AND e OR.	LD: non richiesta AND o OR: richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
FLOATING-POINT TO ASCII (solo CS1-H, CJ1-H, CJ1M o CS1D) FSTR @FSTR 448	FSTR(448) S C D **S:** primo canale sorgente **C:** canale di controllo **D:** canale di destinazione	Converte i dati a virgola mobile in singola precisione specificati (formato decimale o esponenziale a 32 bit) in stringhe di testo (ASCII) e invia il risultato al canale di destinazione.	Uscita richiesta
ASCII TO FLOATING-POINT (solo CS1-H, CJ1-H, CJ1M o CS1D) FVAL @FVAL 449	FVAL(449) S D **S:** canale sorgente **D:** primo canale di destinazione	Converte la rappresentazione in stringhe di testo specificata (ASCII) dei dati a virgola mobile in singola precisione (formato decimale o esponenziale) in dati a virgola mobile in singola precisione a 32 bit e invia il risultato ai canali di destinazione.	Uscita richiesta

3-14 Istruzioni a virgola mobile in doppia precisione

Le istruzioni a virgola mobile in doppia precisione sono supportate solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE FLOATING TO 16-BIT BINARY FIXD @FIXD 841	FIXD(841) S D **S:** primo canale sorgente **D:** canale di destinazione	Converte i dati a virgola mobile in doppia precisione specificati (64 bit) in dati binari a 16 bit con segno e invia il risultato al canale di destinazione.	Uscita Richiesta
DOUBLE FLOATING TO 32-BIT BINARY FIXLD @FIXLD 842	FIXLD(842) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Converte i dati a virgola mobile in doppia precisione specificati (64 bit) in dati binari a 32 bit con segno e invia il risultato ai canali di destinazione.	Uscita Richiesta
16-BIT BINARY TO DOUBLE FLOATING DBL @DBL 843	DBL(843) S D **S:** canale sorgente **D:** primo canale di destinazione	Converte i dati binari a 16 bit con segno specificati in dati a virgola mobile in doppia precisione (64 bit) e invia il risultato ai canali di destinazione.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
32-BIT BINARY TO DOUBLE FLOATING DBLL @DBLL 844	DBLL(844) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Converte i dati binari a 32 bit con segno specificati in dati a virgola mobile in doppia precisione (64 bit) e invia il risultato ai canali di destinazione.	Uscita Richiesta
DOUBLE FLOATING-POINT ADD +D @+D 845	+D(845) Au Ad R **Au:** primo canale addendo **Ad:** primo canale addendo **R:** primo canale del risultato	Aggiunge i valori a virgola mobile in doppia precisione specificati (64 bit ciascuno) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE FLOATING-POINT SUBTRACT –D @–D 846	–D(846) Mi Su R **Mi:** primo canale minuendo **Su:** primo canale sottraendo **R:** primo canale del risultato	Sottrae i valori a virgola mobile in doppia precisione specificati (64 bit ciascuno) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE FLOATING-POINT MULTIPLY *D @*D 847	*D(847) Md Mr R **Md:** primo canale moltiplicando **Mr:** primo canale moltiplicatore **R:** primo canale del risultato	Moltiplica i valori a virgola mobile in doppia precisione specificati (64 bit ciascuno) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE FLOATING-POINT DIVIDE /D @/D 848	/D(848) Dd Dr R **Dd:** primo canale dividendo **Dr:** primo canale divisore **R:** primo canale del risultato	Divide i valori a virgola mobile in doppia precisione specificati (64 bit ciascuno) e invia il risultato ai canali dei risultati.	Uscita Richiesta

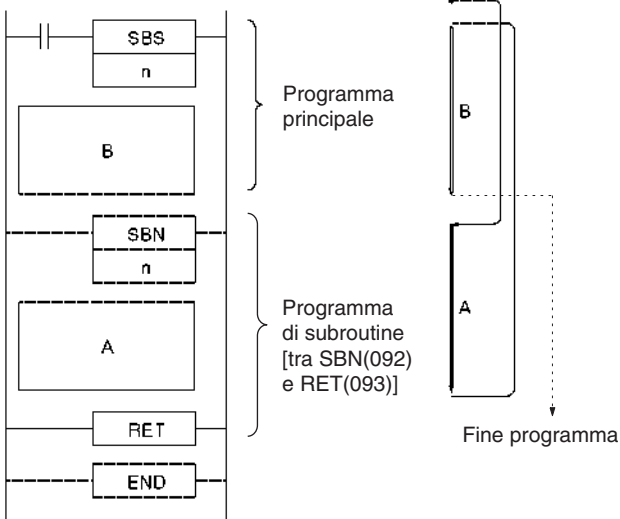
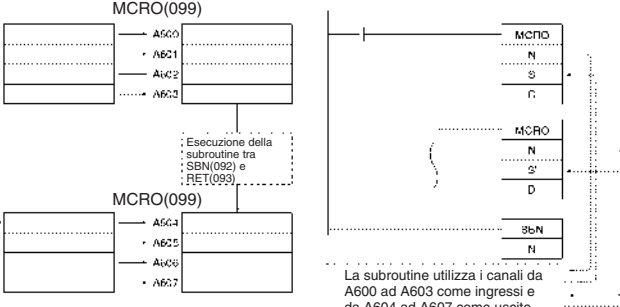
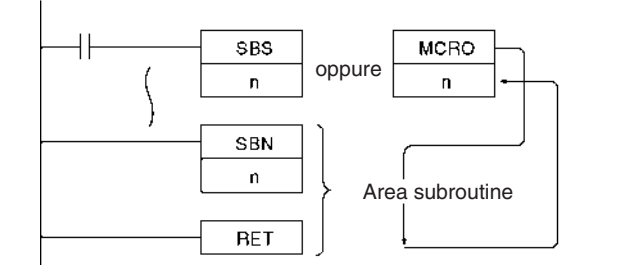
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DOUBLE DEGREES TO RADIANS RADD @RADD 849	RADD(849) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte i dati a virgola mobile in doppia precisione specificati (64 bit) da gradi in radianti e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE RADIANS TO DEGREES DEGD @DEGD 850	DEGD(850) S R **S:** primo canale sorgente **R:** primo canale del risultato	Converte i dati a virgola mobile in doppia precisione specificati (64 bit) da radianti in gradi e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE SINE SIND @SIND 851	SIND(851) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il seno dell'angolo (in radianti) dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE COSINE COSD @COSD 852	COSD(852) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il coseno dell'angolo (in radianti) dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE TANGENT TAND @TAND 853	TAND(853) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola la tangente dell'angolo (in radianti) dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE ARC SINE ASIND @ASIND 854	ASIND(854) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola l'angolo (in radianti) derivandolo dal valore di seno dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati. La funzione di arcoseno è l'inverso della funzione di seno e restituisce l'angolo che produce un dato valore di seno compreso tra -1 e 1.	Uscita Richiesta
DOUBLE ARC COSINE ACOSD @ACOSD 855	ACOSD(855) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola l'angolo (in radianti) derivandolo dal valore di coseno dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati. La funzione di arcocoseno è l'inverso della funzione di coseno e restituisce l'angolo che produce un dato valore di coseno compreso tra -1 e 1.	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
DOUBLE ARC TANGENT ATAND @ATAND 856	ATAND(856) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola l'angolo (in radianti) derivandolo dal valore di tangente dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati. La funzione di arcotangente è l'inverso della funzione di tangente e restituisce l'angolo che produce un dato valore di tangente.	Uscita Richiesta
DOUBLE SQUARE ROOT SQRTD @SQRTD 857	SQRTD(857) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola la radice quadrata dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE EXPONENT EXPD @EXPD 858	EXPD(858) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il valore esponenziale naturale (in base e) dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE LOGARITHM LOGD @LOGD 859	LOGD(859) S R **S:** primo canale sorgente **R:** primo canale del risultato	Calcola il logaritmo naturale (in base e) dei dati a virgola mobile in doppia precisione specificati (64 bit) e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE EXPONENTIAL POWER PWRD @PWRD 860	PWRD(860) B E R **B:** primo canale base **E:** primo canale esponente **R:** primo canale del risultato	Eleva un numero a virgola mobile in doppia precisione (64 bit) alla potenza di un altro numero a virgola mobile in doppia precisione e invia il risultato ai canali dei risultati.	Uscita Richiesta
DOUBLE SYMBOL COMPARISON LD, AND o OR + =D (335), <>D (336), <D (337), <=D (338), >D (339) o >=D (340)	Uso di LD: Simbolo, opzione S1 S2 Uso di AND: Simbolo, opzione S1 S2 Uso di OR: Simbolo, opzione S1 S2 **S1:** dati di confronto 1 **S2:** dati di confronto 2	Confronta i dati in doppia precisione specificati (64 bit) e crea una condizione di esecuzione ON se il risultato del confronto è vero. Con le istruzioni di confronto con simboli a virgola mobile è possibile utilizzare tre tipi di simboli: LD (LOAD), AND e OR.	LD: non richiesta AND o OR: richiesta

3-15 Istruzioni di elaborazione dei dati delle tabelle

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SET STACK SSET @SSET 630	SSET(630) TB N **TB:** primo indirizzo stack **N:** Numero di canali	Definisce uno stack della lunghezza specificata che inizia dal canale specificato e inizializza i canali in tale area dati impostandoli su zero.	Uscita Richiesta
PUSH ONTO STACK PUSH @PUSH 632	PUSH(632) TB S **TB:** primo indirizzo stack **S:** canale sorgente	Scrive un canale di dati nello stack specificato.	Uscita Richiesta
LAST IN FIRST OUT LIFO @LIFO 634	LIFO(634) TB D **TB:** primo indirizzo stack **D:** Canale di destinazione	Legge l'ultimo canale dei dati scritti nello stack specificato, ossia i dati nello stack più recenti.	Uscita Richiesta
FIRST IN FIRST OUT FIFO @FIFO 633	FIFO(633) TB D **TB:** primo indirizzo stack **D:** Canale di destinazione	Legge il primo canale dei dati scritti nello stack specificato, ossia i dati nello stack meno recenti.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DIMENSION RECORD TABLE DIM @DIM 631	DIM(631) N LR NR TB **N:** numero tabella **LR:** lunghezza di ciascun record **NR:** Numero di record **TB:** primo canale tabella	Definisce una tabella di record dichiarando la lunghezza di ciascun record e il numero di record. È possibile definire 16 tabelle di record al massimo.	Uscita Richiesta
SET RECORD LOCATION SETR @SETR 635	SETR(635) N R D **N:** numero tabella **R:** numero record **D:** registro indice di destinazione	Scrive la posizione del record specificato (l'indirizzo di memoria I/O interna relativo all'inizio del record) nel registro indice specificato.	Uscita Richiesta
GET RECORD NUMBER GETR @GETR 636	GETR(636) N IR D **N:** numero tabella **IR:** Registro indice **D:** Canale di destinazione	Restituisce il numero del record all'indirizzo di memoria I/O interna contenuto nel registro indice specificato.	Uscita Richiesta
DATA SEARCH SRCH @SRCH 181	SRCH(181) C R1 Cd **C:** primo canale di controllo **R1:** primo canale nell'intervallo **Cd:** Dati di confronto	Cerca un canale di dati all'interno di un intervallo di canali.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SWAP BYTES SWAP @SWAP 637	SWAP(637) N R1 **N:** Numero di canali **R1:** primo canale nell'intervallo	Scambia i byte più a sinistra e più a destra in tutti i canali dell'intervallo. La posizione del byte viene scambiata. R1 N	Uscita Richiesta
FIND MAXIMUM MAX @MAX 182	MAX(182) C R1 D **C:** primo canale di controllo **R1:** primo canale nell'intervallo **D:** Canale di destinazione	Trova il valore massimo nell'intervallo. Indirizzo di memoria I/O interna R1 C canali A m Valore massimo R1+(W–1) D A IR0 m	Uscita Richiesta
FIND MINIMUM MIN @MIN 183	MIN(183) C R1 D **C:** primo canale di controllo **R1:** primo canale nell'intervallo **D:** Canale di destinazione	Trova il valore minimo nell'intervallo. Indirizzo di memoria I/O interna R1 C canali B m Valore minimo R1+(W–1) D B IR00 m	Uscita Richiesta
SUM SUM @SUM 184	SUM(184) C R1 D **C:** primo canale di controllo **R1:** primo canale nell'intervallo **D:** primo canale di destinazione	Somma i byte o i canali nell'intervallo e invia il risultato a due canali. C W R1 WCH R1+(W–1) +) D+1 D	Uscita Richiesta
FRAME CHECKSUM FCS @FCS 180	FCS(180) C R1 D **C:** primo canale di controllo **R1:** primo canale nell'intervallo **D:** primo canale di destinazione	Calcola il valore FCS (checksum dei frame) ASCII per l'intervallo specificato. R1 C unità Calcolo Conversione ASCII Valore FCS D	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
STACK SIZE READ (solo CS1-H, CJ1-H, CJ1M o CS1D) SNUM @SNUM 638	SNUM(638) TB D **TB:** primo indirizzo stack **D:** canale di destinazione	Conta la quantità di dati di stack (numero di canali) nello stack specificato.	Uscita richiesta
STACK DATA READ (solo CS1-H, CJ1-H, CJ1M o CS1D) SREAD @SREAD 639	SREAD(639) TB C D **TB:** primo indirizzo stack **C:** valore di offset **D:** canale di destinazione	Legge i dati dall'elemento specificato dello stack. Il valore di offset (numero di elementi prima della posizione corrente del puntatore) indica la posizione dell'elemento desiderato.	Uscita richiesta
STACK DATA OVERWRITE (solo CS1-H, CJ1-H, CJ1M o CS1D) SWRIT @SWRIT 640	SWRIT(640) TB C S **TB:** primo indirizzo stack **C:** valore di offset **S:** Dati sorgente	Scrive i dati di origine nell'elemento specificato dello stack, sovrascrivendo i dati già esistenti. Il valore di offset (numero di elementi prima della posizione corrente del puntatore) indica la posizione dell'elemento desiderato.	Uscita richiesta
STACK DATA INSERT (solo CS1-H, CJ1-H, CJ1M o CS1D) SINS @SINS 641	SINS(641) TB C S **TB:** primo indirizzo stack **C:** valore di offset **S:** Dati sorgente	Inserisce i dati sorgente nell'elemento di dati specificato nello stack e fa scorrere verso il basso i dati rimanenti dello stack. Il valore di offset indica la posizione del punto d'inserimento (il numero di elementi di dati che precede la posizione corrente del puntatore).	Uscita richiesta
STACK DATA DELETE (solo CS1-H, CJ1-H, CJ1M o CS1D) SDEL @SDEL 642	SDEL(642) TB C D **TB:** primo indirizzo stack **C:** valore di offset **D:** canale di destinazione	Elimina l'elemento di dati che si trova nella posizione specificata nello stack e fa scorrere verso l'alto i dati rimanenti dello stack. Il valore di offset (numero di elementi prima della posizione corrente del puntatore) indica la posizione del punto di eliminazione.	Uscita richiesta

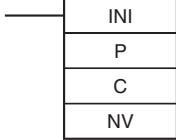
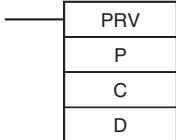
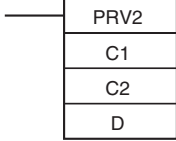
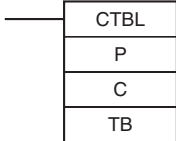
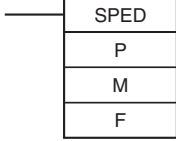
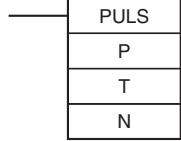
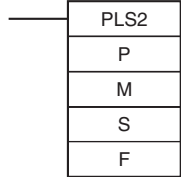
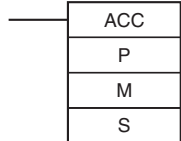
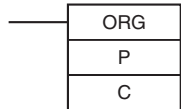
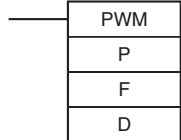
3-16 Istruzioni di controllo dei dati

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
PID CONTROL PID 190	PID(190) S C D **S:** canale di ingresso **C:** primo canale parametro **D:** canale di uscita	Esegue un controllo PID in base ai parametri specificati.	Uscita Richiesta
PID CONTROL WITH AUTOTUNING PIDAT 191 (solo CS1-H, CJ1-H o CJ1M)	PIDAT(191) S C D **S:** canale di ingresso **C:** primo canale parametro **D:** canale di uscita	Esegue un controllo PID in base ai parametri specificati. PIDAT(191) consente di regolare automaticamente le costanti PID.	Uscita richiesta
LIMIT CONTROL LMT @LMT 680	LMT(680) S C D **S:** canale di ingresso **C:** primo canale limite **D:** canale di uscita	Controlla i dati di uscita in base all'appartenenza dei dati di ingresso alla gamma delimitata dai limiti superiore e inferiore.	Uscita Richiesta
DEAD BAND CONTROL BAND @BAND 681	BAND(681) S C D **S:** canale di ingresso **C:** primo canale limite **D:** canale di uscita	Controlla i dati di uscita in base all'appartenenza dei dati di ingresso all'intervallo della zona morta.	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
DEAD ZONE CONTROL ZONE @ZONE 682	ZONE(682) / S / C / D **S:** canale di ingresso **C:** primo canale limite **D:** canale di uscita	Aggiunge la polarizzazione specificata ai dati di ingresso e restituisce il risultato. Uscita Polarizzazione positiva (C+1) Segnale di ingresso 0 Polarizzazione negativa (C)	Uscita Richiesta
TIME-PROPORTIONAL OUTPUT TPO 685 (solo CPU della serie CS/CJ versione 2.0 o successiva)	TPO(685) / S / C / R **S:** canale di ingresso **C:** primo canale parametro **R:** bit di uscita a impulsi	A partire dal duty-cycle o da una variabile manipolata contenuta nel canale specificato, converte il duty-cycle in un'uscita proporzionale al tempo in base ai parametri specificati ed emette il risultato dall'uscita specificata.	Uscita Richiesta
SCALING SCL @SCL 194	SCL(194) / S / P1 / R **S:** canale sorgente **P1:** primo canale parametro **R:** canale del risultato	Converte dati binari senza segno in dati in formato BCD senza segno in base alla funzione lineare specificata. R (BCD senza segno) La funzione di scala viene eseguita in base alla funzione lineare definita dai punti A e B. Punto B Punto A Bd Ad As Bs S (binario senza segno) P: Ad (BCD) P1 + 1: As Binario — Valore convertito P1 +2: Bd (BCD) P1 +3: Bs Binario — Valore convertito	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SCALING 2 SCL2 @SCL2 486	SCL2(486) S P1 R **S:** canale sorgente **P1:** primo canale parametro **R:** canale del risultato	Converte dati binari con segno in dati in formato BCD con segno in base alla funzione lineare specificata. È possibile specificare un offset nella definizione della funzione lineare.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SCALING 3 SCL3 @SCL3 487	SCL3(487) S P1 R **S:** canale sorgente **P1:** primo canale parametro **R:** canale del risultato	Converte dati in formato BCD con segno in dati binari con segno in base alla funzione lineare specificata. È possibile specificare un offset nella definizione della funzione lineare.	Uscita Richiesta
AVERAGE AVG 195	AVG(195) S N R **S:** canale sorgente **N:** numero di cicli **R:** canale del risultato	Calcola il valore medio di un canale di ingresso per il numero di cicli specificato.	Uscita Richiesta

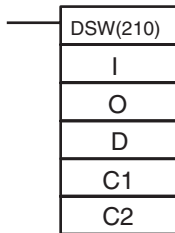
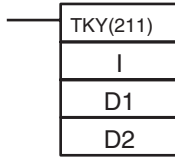
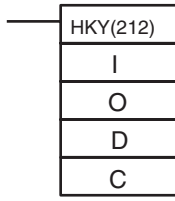
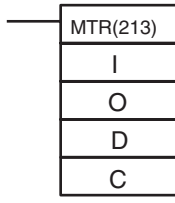
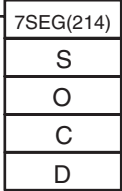
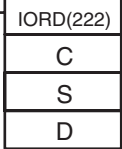
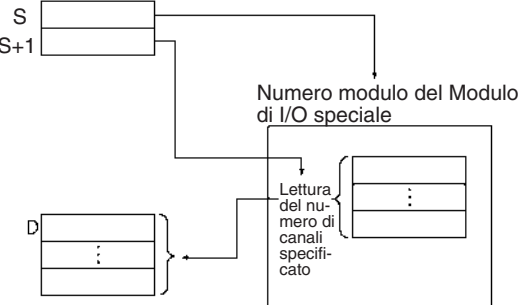
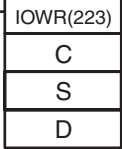
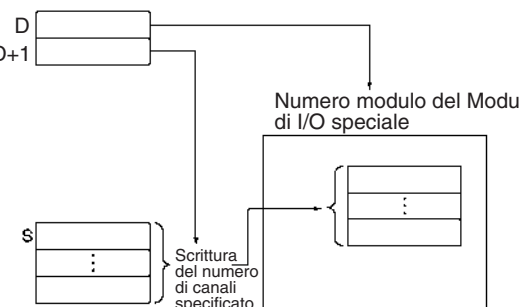
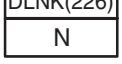
3-17 Istruzioni di subroutine

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SUBROUTINE CALL SBS @SBS 091	SBS(091) N **N:** N. subroutine	Richiama la subroutine con il numero di subroutine specificato ed esegue il programma.	Uscita Richiesta
MACRO MCRO @MCRO 099	MCRO(099) N S D **N:** N. subroutine **S:** primo canale parametro di ingresso **D:** primo canale parametro di uscita	Chiama la subroutine con il numero specificato ed esegue il programmautilizzando i parametri di ingresso specificati da S a S+3 e i parametri di uscita specificati da D a D+3.	Uscita Richiesta
SUBROUTINE ENTRY SBN 092	SBN(092) N **N:** N. subroutine	Indica l'inizio del programma di subroutine con il numero di subroutine specificato.	Uscita Non richiesta
SUBROUTINE RETURN RET 093	RET(093)	Indica la fine del programma di subroutine.	Uscita Non richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
GLOBAL SUBROUTINE CALL (solo CS1-H, CJ1-H, CJ1M o CS1D) GSBS 750	GSBS(750) N **N:** numero subroutine	Richiama la subroutine con il numero di subroutine specificato ed esegue il programma.	Uscita Non richiesta
GLOBAL SUBROUTINE ENTRY (solo CS1-H, CJ1-H, CJ1M o CS1D) GSBN 751	GSBN(751) N **N:** numero subroutine	Indica l'inizio del programma di subroutine con il numero di subroutine specificato.	Uscita Non richiesta
GLOBAL SUBROUTINE RETURN (solo CS1-H, CJ1-H, CJ1M o CS1D) GRET 752	GRET(752)	Indica la fine del programma di subroutine.	Uscita Non richiesta

3-18 Istruzioni di controllo degli interrupt

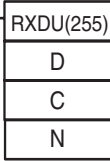
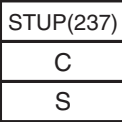
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SET INTERRUPT MASK (non supportata dalle CPU CS1D per sistemi a due CPU). MSKS @MSKS 690	MSKS(690) N S **N:** ID interrupt **S:** dati interrupt	Imposta l'elaborazione degli interrupt per gli interrupt di I/O e programmati. Alla prima accensione del PLC, sia i task ad interrupt di I/O che programmati sono mascherati (disabilitati). Per smascherare e mascherare gli interrupt di I/O e per impostare gli intervalli di tempo per gli interrupt programmati, è possibile usare l'istruzione MSKS(690). Modulo di interrupt di ingresso da 0 a 3 Interrupt di I/O Ingressi di interrupt da 0 a 7 mascherati (1) o smascherati (0). Intervallo di tempo Interrupt programmato Imposta l'intervallo di tempo dell'interrupt programmato.	Uscita Richiesta
READ INTERRUPT MASK (non supportata dalle CPU CS1D per sistemi a due CPU). MSKR @MSKR 692	MSKR(692) N D **N:** ID interrupt **D:** Canale di destinazione	Legge le impostazioni di elaborazione dell'interrupt correnti definite tramite l'istruzione MSKS(690).	Uscita Richiesta

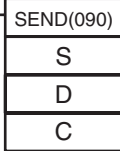
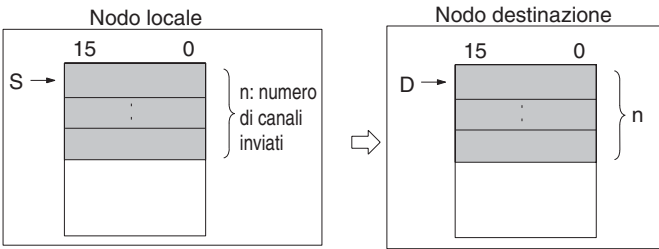
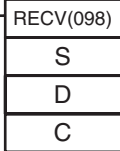
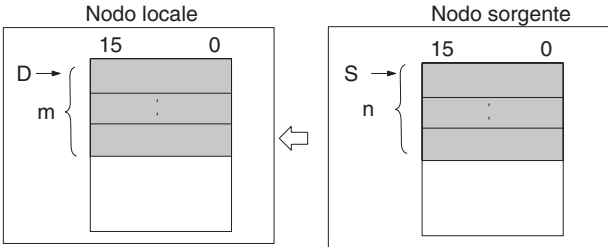
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
CLEAR INTERRUPT (non supportata dalle CPU CS1D per sistemi a due CPU). CLI @CLI 691	CLI(691) N S **N:** ID interrupt **S:** dati interrupt	Cancella o mantiene gli ingressi di interrupt registrati per gli interrupt di I/O o imposta il tempo del primo interrupt per gli interrupt programmati.	Uscita Richiesta
DISABLE INTERRUPTS DI @DI 693	Istruzione DI(693)	Disabilita l'esecuzione dei task ad interrupt eccetto quello di spegnimento.	Uscita Richiesta
ENABLE INTERRUPTS EI 694	Istruzione EI(694)	Abilita l'esecuzione di tutti i task ad interrupt che erano stati disabilitati tramite l'istruzione DI(693).	Uscita Non richiesta

3-19 Istruzioni per contatore veloce e uscita a impulsi (solo CJ1M-CPU21/22/23)

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
MODE CONTROL INI @INI 880	INI P C NV **P:** identificatore della porta **C:** dati di controllo **NV:** primo canale con nuovo valore attuale	L'istruzione INI(880) viene utilizzata per avviare e interrompere il confronto dei valori di riferimento, modificare il valore attuale di un contatore veloce, modificare il valore attuale di un interrupt di ingresso (modalità contatore), modificare il valore attuale di un'uscita a impulsi o arrestare l'uscita.	Uscita Richiesta
HIGH-SPEED COUNTER PV READ PRV @PRV 881	PRV P C D **P:** identificatore della porta **C:** dati di controllo **D:** primo canale di destinazione	L'istruzione PRV(881) viene utilizzata per leggere il valore attuale di un contatore veloce, di un'uscita a impulsi o di un interrupt di ingresso (modalità contatore).	Uscita Richiesta
COUNTER FREQUENCY CONVERT PRV2 883 (solo CPU CJ1M versione 2.0 o successiva)	PRV2 C1 C2 D **C1:** dati di controllo **C2:** impulsi/giro **D:** primo canale di destinazione	Legge l'ingresso della frequenza a impulsi di un contatore veloce e converte la frequenza in velocità di rotazione (numero di rotazioni) oppure converte il valore attuale del contatore nel numero totale di rotazioni. Il risultato viene inviato ai canali di destinazione in formato esadecimale a 8 cifre. Gli impulsi possono essere immessi esclusivamente dal contatore veloce 0.	Uscita Richiesta
COMPARISON TABLE LOAD CTBL @CTBL 882	CTBL P C TB **P:** identificatore della porta **C:** dati di controllo **TB:** primo canale della tabella di confronto	L'istruzione CTBL(882) viene utilizzata per eseguire confronti tra il valore attuale di un contatore veloce e valori o intervalli di riferimento.	Uscita Richiesta
SPEED OUTPUT SPED @SPED 885	SPED P M F **P:** identificatore della porta **M:** modalità di uscita **F:** primo canale di frequenza degli impulsi	L'istruzione SPED(885) viene utilizzata per specificare la frequenza e consentire l'emissione degli impulsi senza accelerazione o decelerazione.	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
SET PULSES PULS @PULS 886	PULS P T N **P:** identificatore della porta **T:** tipo di impulsi **N:** numero di impulsi	L'istruzione PULS(886) viene utilizzata per impostare il numero di impulsi per l'uscita a impulsi.	Uscita Richiesta
PULSE OUTPUT PLS2 @PLS2 887	PLS2 P M S F **P:** identificatore della porta **M:** modalità di uscita **S:** primo canale della tabella delle impostazioni **F:** primo canale della frequenza iniziale	L'istruzione PLS2(887) viene utilizzata per impostare la frequenza degli impulsi e i valori di accelerazione e decelerazione, nonché consentire l'emissione degli impulsi con accelerazione o decelerazione (con valori di accelerazione e decelerazione distinti). È previsto solo il posizionamento.	Uscita Richiesta
ACCELERATION CONTROL ACC @ACC 888	ACC P M S **P:** identificatore della porta **M:** modalità di uscita **S:** primo canale della tabella delle impostazioni	L'istruzione ACC(888) viene utilizzata per impostare la frequenza degli impulsi e i valori di accelerazione e decelerazione, nonché consentire l'emissione degli impulsi con accelerazione o decelerazione (con valori di accelerazione e decelerazione uguali). Sono previsti sia il posizionamento che il controllo della velocità.	Uscita Richiesta
ORIGIN SEARCH ORG @ORG 889	ORG P C **P:** identificatore della porta **C:** dati di controllo	L'istruzione ORG(889) viene utilizzata per eseguire operazioni di ricerca dell'origine e ritorno all'origine.	Uscita Richiesta
PULSE WITH VARIABLE DUTY FACTOR PWM @ 891	PWM P F D **P:** identificatore della porta **F:** frequenza **D:** duty-cycle	L'istruzione PWM(891) viene utilizzata per l'emissione di impulsi con duty-cycle variabile.	Uscita Richiesta

3-20 Istruzioni di step

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
STEP DEFINE STEP 008	STEP(008) B **B:** Bit	STEP(008) opera nei due modi indicati di seguito, a seconda della posizione e della presenza o meno di un bit di controllo specificato. (1) Avvia uno step specifico. (2) Termina l'area della programmazione step (ovvero l'esecuzione step).	Uscita Richiesta
STEP START SNXT 009	SNXT(009) B **B:** Bit	L'istruzione SNXT(009) viene utilizzata nei seguenti modi: (1) Per avviare l'esecuzione della programmazione step. (2) Per passare al bit di controllo dello step successivo. (3) Per terminare l'esecuzione della programmazione step.	Uscita Richiesta

3-21 Istruzioni per Moduli di I/O di base

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
I/O REFRESH (aggiornamento degli I/O) IORF @IORF 097	IORF(097) St E **St:** canale di inizio **E:** canale di fine	Aggiorna i canali di I/O specificati.	Uscita Richiesta
7-SEGMENT DECODER SDEC @SDEC 078	SDEC(078) S Di D **S:** canale sorgente **Di:** identificatore della cifra **D:** primo canale di destinazione	Converte il contenuto esadecimale delle cifre specificate in codici display a 7 segmenti e li memorizza negli 8 bit superiori o inferiori dei canali di destinazione specificati.	Uscita Richiesta

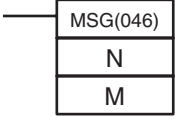
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DIGITAL SWITCH INPUT DSW 210 (solo CPU della serie CS/CJ versione 2.0 o successiva)	DSW(210) I O D C1 C2 **I :** canale di ingresso dati (da D0 a D3) **O :** canale di uscita **D:** primo canale del risultato **C1:** Numero di cifre **C2:** canale di sistema	Legge il valore impostato su un selettore digitale esterno (o selettore rotativo) collegato a un Modulo di ingresso o di uscita e memorizza i dati in formato BCD a 4 o 8 cifre nei canali specificati.	Uscita Richiesta
TEN KEY INPUT TKY 211 (solo CPU della serie CS/CJ versione 2.0 o successiva)	TKY(211) I D1 D2 **I:** canale di ingresso dati **D1:** primo canale registro **D2:** canale di ingresso tasto	Legge i dati numerici da un tastierino a dieci tasti collegato a un Modulo di ingresso e memorizza fino a 8 cifre di dati in formato BCD nei canali specificati.	Uscita Richiesta
HEXADECIMAL KEY INPUT HKY 212 (solo CPU della serie CS/CJ versione 2.0 o successiva)	HKY(212) I O D C **I:** canale di ingresso dati **O:** canale di uscita **D:** primo canale registro **C:** canale di sistema	Legge i dati numerici da un tastierino esadecimale collegato a un Modulo di ingresso e a un Modulo di uscita e memorizza fino a 8 cifre di dati in formato esadecimale nei canali specificati.	Uscita Richiesta
MATRIX INPUT MTR 213 (solo CPU della serie CS/CJ versione 2.0 o successiva)	MTR(213) I O D C **I:** canale di ingresso dati **O:** canale di uscita **D:** primo canale di destina-zione **C:** canale di sistema	Accetta fino a 64 segnali da una matrice 8 × 8 collegata a un Modulo di ingresso e a un Modulo di uscita (utilizzando 8 punti di ingresso e 8 punti di uscita) e memorizza i dati a 64 bit nei 4 canali di destinazione.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
7-SEGMENT DISPLAY OUTPUT 7SEG 214 (solo CPU della serie CS/CJ versione 2.0 o successiva)	7SEG(214) S O C D **S:** primo canale sorgente **O:** canale di uscita **C:** dati di controllo **D:** canale di sistema	Converte i dati sorgente (a 4 o 8 cifre in formato BCD) nel formato per il display a 7 segmenti, quindi emette tali dati sul canale di uscita specificato.	Uscita Richiesta
INTELLIGENT I/O READ IORD @IORD 222	IORD(222) C S D **C:** dati di controllo **S:** numero di canali e sorgente per il trasferimento **D:** numero di canali e destinazione per il trasferimento	Legge il contenuto dell'area di memoria del Modulo di I/O speciale o del Modulo CPU bus (vedere nota). S S+1 Numero modulo del Modulo di I/O speciale Lettura del numero di canali specificato D **Nota:** le CPU della serie CS/CJ versione 2.0 o successiva (comprese le CPU CS1-H, CJ1-H e CJ1M con numero di lotto 030418 o successivo) possono leggere dai Moduli CPU bus.	Uscita Richiesta
INTELLIGENT I/O WRITE IOWR @IOWR 223	IOWR(223) C S D **C:** dati di controllo **S:** numero di canali e sorgente per il trasferimento **D:** numero di canali e destinazione per il trasferimento	Invia il contenuto dell'area di memoria I/O della CPU al Modulo di I/O speciale o al Modulo CPU bus (vedere nota). D D+1 Numero modulo del Modulo di I/O speciale S Scrittura del numero di canali specificato **Nota:** le CPU della serie CS/CJ versione 2.0 o successiva (comprese le CPU CS1-H, CJ1-H e CJ1M con numero di lotto 030418 o successivo) possono scrivere nei Moduli CPU bus.	Uscita Richiesta
CPU BUS UNIT I/O REFRESH (solo CS1-H, CJ1-H, CJ1M o CS1D) DLNK @DLNK 226	DLNK(226) N **N:** numero modulo	Esegue immediatamente l'aggiornamento degli I/O del Modulo CPU bus con il numero di modulo specificato.	Uscita richiesta

3-22 Istruzioni per la comunicazione seriale

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
PROTOCOL MACRO PMCR @PMCR 260	PMCR(260) C1 C2 S R **C1:** canale di controllo 1 **C2:** canale di controllo 2 **S:** primo canale inviato **R:** primo canale ricevuto	Richiama ed esegue una sequenza di comunicazione registrata in un Modulo di comunicazione seriale o in una Scheda di comunicazione seriale (solo serie CS). CPU / Modulo di comunicazione seriale / Porta / S n a n / R m a / Dispositivo esterno	Uscita Richiesta
TRANSMIT TXD @TXD 236	TXD(236) S C N **S:** primo canale sorgente **C:** canale di controllo **N:** numero di byte Da 0000 a 0100 esadecimale (da 0 a 256 decimale)	Invia il numero di byte di dati specificato dalla porta RS-232C integrata della CPU.	Uscita Richiesta
RECEIVE RXD @RXD 235	RXD(235) D C N **D:** primo canale di destinazione **C:** canale di controllo **N:** numero di byte da memorizzare da 0000 a 0100 esadecimale (da 0 a 256 decimale)	Legge il numero di byte di dati specificato dalla porta RS-232C integrata della CPU.	Uscita Richiesta
TRANSMIT VIA SERIAL COMMU-NICATIONS UNIT TXDU @TXDU 256	TXDU(256) S C N **S**: primo canale sorgente **C** - primo canale di controllo **N**: numero di byte (0000 … 0256 BCD)	Invia il numero di byte di dati specificato, senza applicare alcuna conversione, dalla porta seriale di un Modulo di comunicazione seriale versione 1.2 o successiva. I dati vengono inviati in modalità nessun protocollo con gli eventuali codici di inizio e di fine specificati nell'area DM allocata.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
RECEIVE VIA SERIAL COMMU-NICATIONS UNIT RXDU @RXDU 255	RXDU(255) D C N **D**: primo canale di destinazione **C**: primo canale di controllo **N**: numero di byte da memorizzare (0000 ... 0100 esadecimale)	Legge il numero di byte di dati specificato a partire dal primo canale specificato dalla porta seriale di un Modulo di comunicazione seriale versione 1.2 o successiva. I dati vengono letti in modalità nessun protocollo con gli eventuali codici di inizio e di fine specificati nell'area di configurazione DM allocata.	
CHANGE SERIAL PORT SETUP STUP @STUP 237	STUP(237) C S **C:** canale di controllo (porta) **S:** Primo canale sorgente	Modifica i parametri di comunicazione di una porta seriale sulla CPU, sul Modulo di comunicazione seriale (Modulo CPU bus) o sulla Scheda di comunicazione seriale. L'istruzione STUP(237) consente pertanto di modificare la modalità di protocollo durante il funzionamento del PLC.	Uscita Richiesta

3-23 Istruzioni di rete

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
NETWORK SEND SEND @SEND 090	SEND(090) S D C **S:** primo canale sorgente **D:** primo canale di destinazione **C:** primo canale di controllo	Trasmette i dati a un nodo nella rete. Nodo locale: 15 0, S →, n: numero di canali inviati Nodo destinazione: 15 0, D →, n	Uscita Richiesta
NETWORK RECEIVE RECV @RECV 098	RECV(098) S D C **S:** primo canale sorgente **D:** primo canale di destinazione **C:** primo canale di controllo	Richiede la trasmissione di dati da un nodo nella rete e riceve tali dati. Nodo locale: 15 0, D →, m Nodo sorgente: 15 0, S →, n	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DELIVER COMMAND CMND @CMND 490	CMND(490) S D C **S:** primo canale del comando **D:** primo canale della risposta **C:** primo canale di controllo	Invia comandi FINS e riceve la risposta Nodo locale: S → 15 … 0; (S−1) + n/2; Dati del comando (n byte); D → 15 … 0; (D−1) + m/2; Dati della risposta (m byte) Comando ⇨ Nodo destinazione: Interpretazione ⇩ Esecuzione; Risposta ⇦	Uscita Richiesta
EXPLICIT MESSAGE SEND EXPLT 720 (solo CPU della serie CS/CJ versione 2.0 o successiva)	EXPLT(720) S D C **S:** primo canale del messaggio di invio **D:** primo canale del messaggio ricevuto **C:** primo canale di controllo	Invia un messaggio esplicito con qualsiasi codice di servizio.	Uscita Richiesta
EXPLICIT GET ATTRIBUTE EGATR 721 (solo CPU della serie CS/CJ versione 2.0 o successiva)	EGATR(721) S D C **S:** primo canale del messaggio di invio **D:** primo canale del messaggio ricevuto **C:** primo canale di controllo del messaggio	Legge le informazioni sullo stato con un messaggio esplicito (Get Attribute Single, codice di servizio: 0E esadecimale).	Uscita Richiesta
EXPLICIT SET ATTRIBUTE ESATR 722 (solo CPU della serie CS/CJ versione 2.0 o successiva)	ESATR(722) S C **S:** primo canale del messaggio di invio **C:** primo canale di controllo	Scrive le informazioni sullo stato con un messaggio esplicito (Set Attribute Single, codice di servizio: 0E esadecimale).	Uscita Richiesta

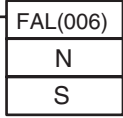
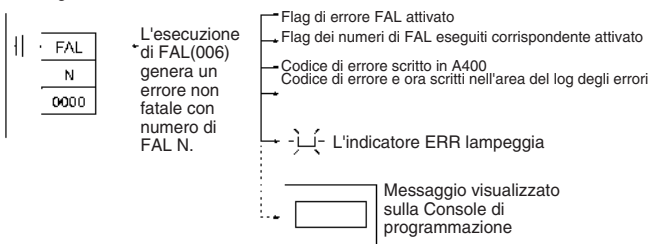
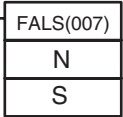
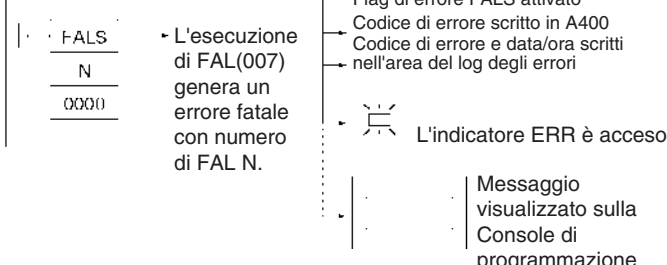
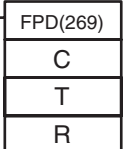
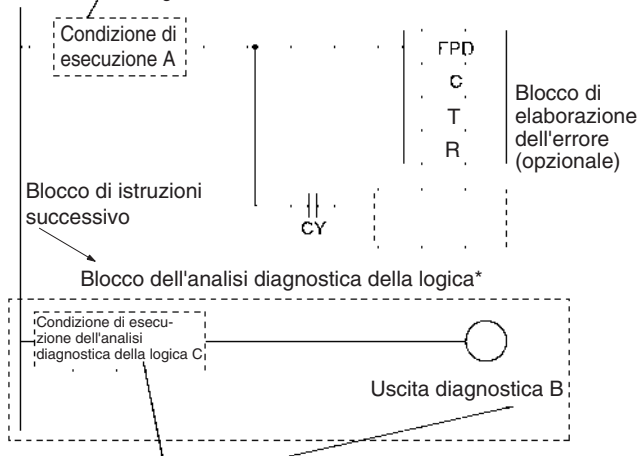
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
EXPLICIT WORD READ ECHRD 723 (solo CPU della serie CS/CJ versione 2.0 o successiva)	ECHRD(723) S D C **S** : primo canale sorgente nella CPU remota **D** : primo canale di destinazione nella CPU locale **C** : primo canale di controllo	Legge dati da una CPU remota in rete sulla CPU locale (la CPU remota deve supportare i messaggi espliciti).	Uscita Richiesta
EXPLICIT WORD WRITE ECHWR 724 (solo CPU della serie CS/CJ versione 2.0 o successiva)	ECHWR(724) S D C **S** : primo canale sorgente nella CPU locale **D** : primo canale di destinazione nella CPU remota **C** : primo canale di controllo	Scrive dati della CPU locale in una CPU remota in rete (la CPU remota deve supportare i messaggi espliciti).	Uscita Richiesta

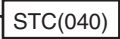
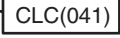
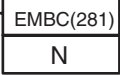
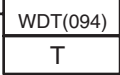
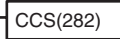
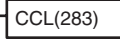
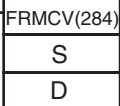
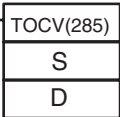
3-24 Istruzioni relative alla memoria file

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
READ DATA FILE FREAD @FREAD 700	**C:** canale di controllo **S1:** primo canale sorgente **S2:** nome file **D:** primo canale di destinazione	Legge la quantità di dati o i dati specificati dal file di dati indicato nella memoria file nella zona dati specificata della CPU.	Uscita Richiesta
WRITE DATA FILE FWRIT @FWRIT 701	**C:** canale di controllo **D1:** primo canale di destinazione **D2:** nome file **S:** primo canale sorgente	Sovrascrive o aggiunge i dati specificati ai dati contenuti nel file di dati indicato nella memoria file dalla zona dati della CPU. Se il file indicato non esiste, viene creato un nuovo file con tale nome.	Uscita Richiesta

3-25 Istruzioni di visualizzazione

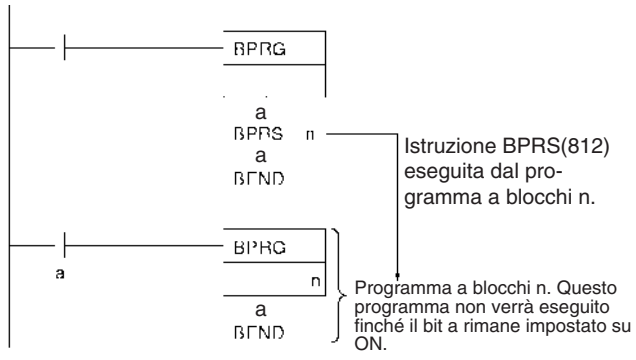
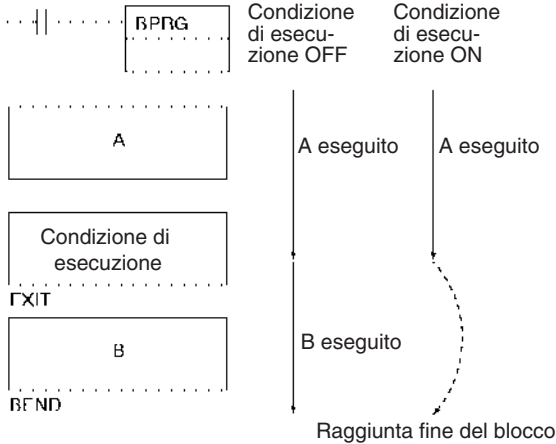
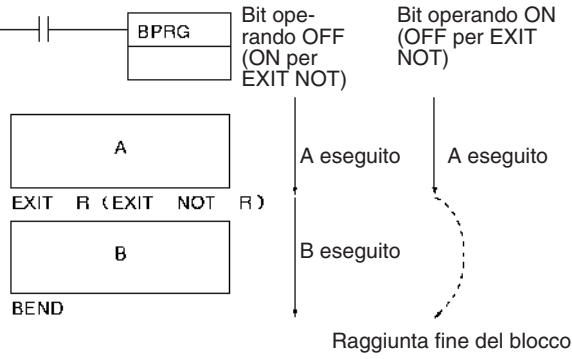
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DISPLAY MESSAGE MSG @MSG 046	MSG(046) N M **N:** numero messaggio **M:** primo canale messaggio	Legge i sedici canali di codice ASCII esteso specificati e visualizza il messaggio su un dispositivo periferico quale una Console di programmazione.	Uscita Richiesta

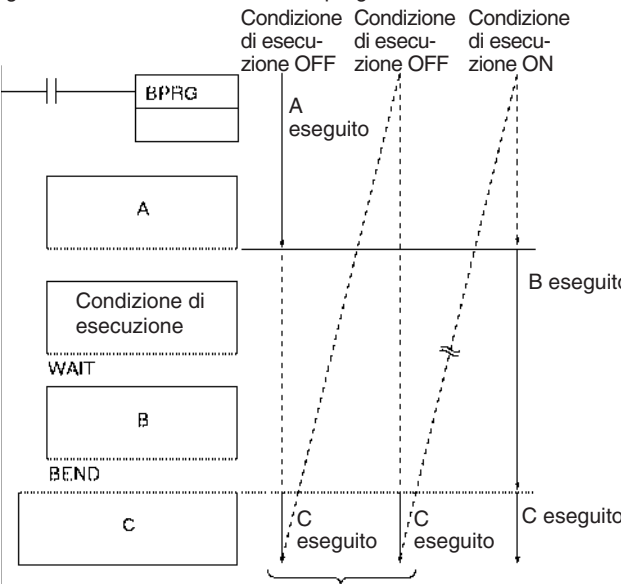
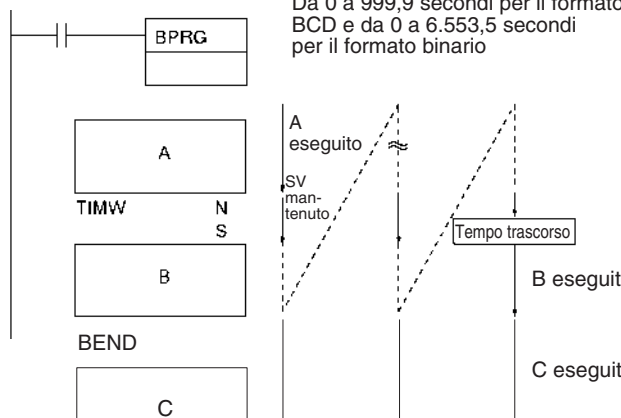
3-26 Istruzioni per l'orologio

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
CALENDAR ADD CADD @CADD 730	CADD(730) C T R **C:** primo canale calendario **T:** primo canale ora **R:** primo canale del risultato	Aggiunge tempo ai dati di calendario nei canali specificati. 15 8 7 0 C: Minuti / Secondi C+1: Giorno / Ora C+2: Anno / Mese + 15 8 7 0 T: Minuti / Secondi T+1: Ore ↓ 15 8 7 0 R: Minuti / Secondi R+1: Giorno / Ora R+2: Anno / Mese	Uscita Richiesta
CALENDAR SUBTRACT CSUB @CSUB 731	CSUB(731) C T R **C:** primo canale calendario **T:** primo canale ora **R:** primo canale del risultato	Sottrae tempo dai dati di calendario nei canali specificati. 15 8 7 0 C: Minuti / Secondi C+1: Giorno / Ora C+2: Anno / Mese 15 8 7 0 T: Minuti / Secondi T+1: Ore ↓ 15 8 7 0 R: Minuti / Secondi R+1: Giorno / Ora R+2: Anno / Mese	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
HOURS TO SECONDS SEC @SEC 065	SEC(065) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Converte i dati temporali in formato ore/minuti/secondi in un tempo equivalente espresso solo in secondi. 15 0 S: Minuti ¦ Secondi S+1: Ore ↓ 15 0 D, D+1: Secondi	Uscita Richiesta
SECONDS TO HOURS HMS @HMS 066	HMS(066) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Converte i dati temporali espressi in secondi in un tempo equivalente in formato ore/minuti/secondi. 15 0 S, S+1: Secondi ↓ 15 0 D: Minuti ¦ Secondi D+1: Ore	Uscita Richiesta
CLOCK ADJUSTMENT DATE @DATE 735	DATE(735) S **S:** primo canale sorgente	Modifica l'impostazione dell'orologio interno in base all'impostazione dei canali sorgente specificati. CPU Clock interno Nuova impostazione S1: Minuti ¦ Secondi S+1: Giorno ¦ Ora S+2: Anno ¦ Mese S+3: 00 ¦ Giorno della settimana	Uscita Richiesta

3-27 Istruzioni di debug

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
TRACE MEMORY SAMPLING TRSM 045	TRSM(045)	Quando viene eseguita l'istruzione TRSM(045), lo stato del bit o del canale preselezionato viene campionato e memorizzato nella memoria di registrazione. È possibile utilizzare TRSM(045) in qualsiasi punto del programma e un numero di volte illimitato.	Uscita Non richiesta

3-28 Istruzioni per la diagnostica di funzionamento incorretto

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
FAILURE ALARM FAL @FAL 006	FAL(006) N S **N:** Numero FAL **S:** primo canale del messaggio o codice di errore da generare	Genera o cancella errori non fatali definiti dall'utente. Gli errori non fatali non causano l'arresto del PLC. Inoltre genera errori non fatali nel sistema. FAL / N / 0000 L'esecuzione di FAL(006) genera un errore non fatale con numero di FAL N. Flag di errore FAL attivato Flag dei numeri di FAL eseguiti corrispondente attivato Codice di errore scritto in A400 Codice di errore e ora scritti nell'area del log degli errori L'indicatore ERR lampeggia Messaggio visualizzato sulla Console di programmazione	Uscita Richiesta
SEVERE FAILURE ALARM FALS 007	FALS(007) N S **N:** numero di FALS **S:** primo canale del messaggio o codice di errore da generare	Genera errori fatali definiti dall'utente. Gli errori fatali causano l'arresto Inoltre genera errori fatali nel sistema. FALS / N / 0000 L'esecuzione di FAL(007) genera un errore fatale con numero di FAL N. Flag di errore FALS attivato Codice di errore scritto in A400 Codice di errore e data/ora scritti nell'area del log degli errori L'indicatore ERR è acceso Messaggio visualizzato sulla Console di programmazione	Uscita Richiesta
FAILURE POINT DETECTION FPD 269	FPD(269) C T R **C:** canale di controllo **T:** tempo di monitoraggio **R:** primo canale registro	L'istruzione FPD(269) consente di rilevare un errore in un blocco di istruzioni monitorando il tempo che intercorre tra l'esecuzione dell'istruzione stessa e l'attivazione di un'uscita di diagnostica, individuando in tal modo l'ingresso che impedisce l'attivazione di un'uscita. Funzione di monitoraggio del tempo Il monitoraggio inizia quando la condizione di esecuzione diventa ON. Se l'uscita B non viene attivata entro il tempo di monitoraggio, viene generato un errore non fatale. Condizione di esecuzione A FPD / C / T / R Blocco di elaborazione dell'errore (opzionale) CY Blocco di istruzioni successivo Blocco dell'analisi diagnostica della logica* Condizione di esecuzione dell'analisi diagnostica della logica C Uscita diagnostica B Funzione di analisi diagnostica della logica Determina quale ingresso in C impedisce l'attivazione dell'uscita B.	Uscita Richiesta

3-29 Altre istruzioni

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
SET CARRY STC @STC 040	STC(040)	Imposta il flag di riporto (CY).	Uscita Richiesta
CLEAR CARRY CLC @CLC 041	CLC(041)	Disattiva il flag di riporto (CY).	Uscita Richiesta
SELECT EM BANK EMBC @EMBC 281	EMBC(281) N **N:** Numero del banco EM	Modifica il banco EM corrente.	Uscita Richiesta
EXTEND MAXIMUM CYCLE TIME WDT @WDT 094	WDT(094) T **T:** impostazione temporizzatore	Estende il tempo di ciclo massimo solo per il ciclo in cui viene eseguita l'istruzione.	Uscita Richiesta
SAVE CONDITION FLAGS (solo CS1-H, CJ1-H, CJ1M o CS1D) CCS @CCS 282	CCS(282)	Salva lo stato dei flag di condizione.	Uscita Richiesta
LOAD CONDITION FLAGS (solo CS1-H, CJ1-H, CJ1M o CS1D) CCL @CCL 283	CCL(283)	Legge lo stato dei flag di condizione precedentemente salvato.	Uscita Richiesta
CONVERT ADDRESS FROM CV (solo CS1-H, CJ1-H, CJ1M o CS1D) FRMCV @FRMCV 284	FRMCV(284) S D **S:** canale contenente l'indirizzo di memoria della serie CV **D:** registro indice di destinazione	Converte l'indirizzo di memoria di un PLC della serie CV nell'indirizzo di memoria equivalente di un PLC della serie CS/CJ.	Uscita Richiesta
CONVERT ADDRESS TO CV (solo CS1-H, CJ1-H, CJ1M o CS1D) TOCV @TOCV 285	TOCV(285) S D **S:** registro indice contenente l'indirizzo di memoria della serie CS **D:** canale di destinazione	Converte l'indirizzo di memoria di un PLC della serie CS/CJ nell'indirizzo di memoria equivalente di un PLC della serie CV.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
DISABLE PERIPHERAL SERVICING (solo CPU CS1D per sistemi a singola CPU, CS1-H, CJ1-H o CJ1M) IOSP @IOSP 287	IOSP(287)	Disabilita la gestione delle periferiche durante l'esecuzione del programma in una delle modalità di elaborazione parallela o in modalità di gestione prioritaria delle periferiche.	Uscita Richiesta
ENABLE PERIPHERAL SERVICING (solo CPU CS1D per sistemi a singola CPU, CS1-H, CJ1-H o CJ1M) IORS 288	IORS(288)	Abilita la gestione delle periferiche disabilitata tramite l'istruzione IOSP(287) per l'esecuzione del programma in una delle modalità di elaborazione parallela o in modalità di gestione prioritaria delle periferiche.	Uscita Non richiesta

3-30 Istruzioni di programmazione a blocchi

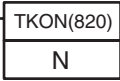
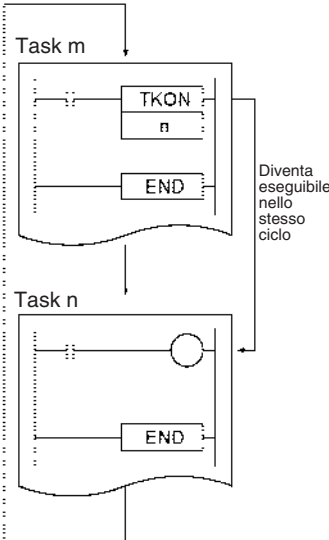
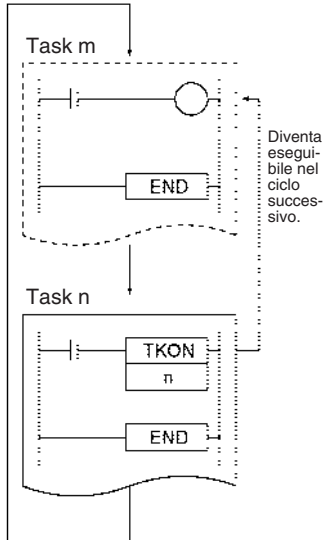
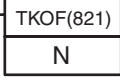
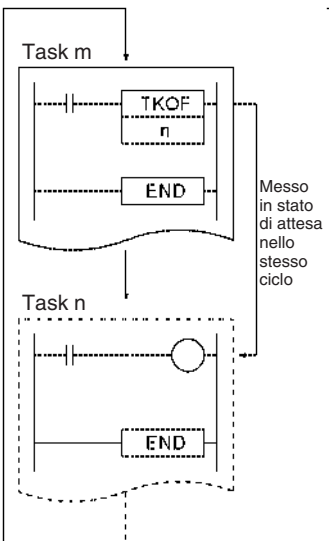
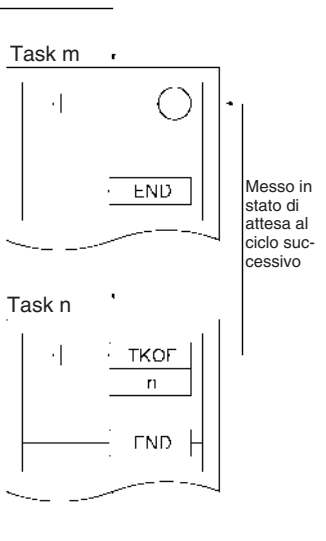
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzioni	Posizione Condizione di esecuzione
BLOCK PROGRAM BEGIN BPRG 096	BPRG(096) N **N:** numero del programma a blocchi	Definisce un'area di programmazione a blocchi. Per ogni istruzione BPRG(096) deve esistere un'istruzione BEND(801) corrispondente.	Uscita Richiesta
BLOCK PROGRAM END BEND 801		Definisce un'area di programmazione a blocchi. Per ogni istruzione BPRG(096) deve esistere un'istruzione BEND(801) corrispondente.	Programma a blocchi Richiesta
BLOCK PROGRAM PAUSE BPPS 811	BPPS (811) N **N:** numero del programma a blocchi	Mette in pausa e riavvia il programma a blocchi specificato da un altro programma a blocchi.	Programma a blocchi Richiesta

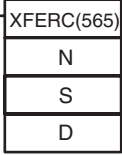
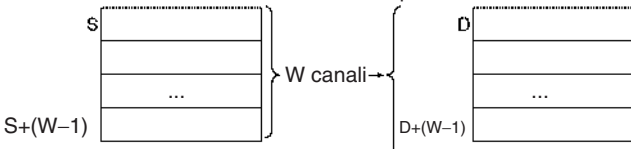
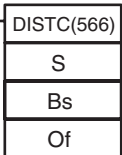
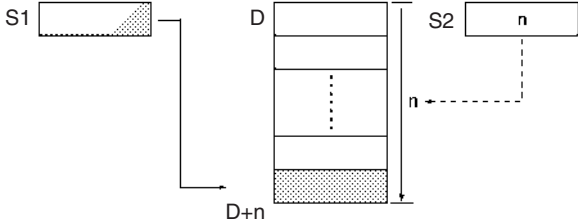
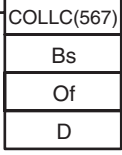
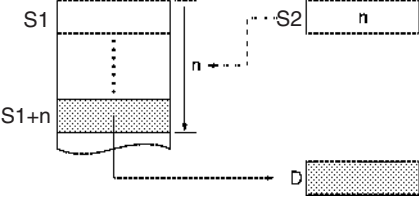
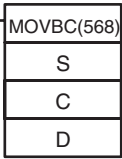
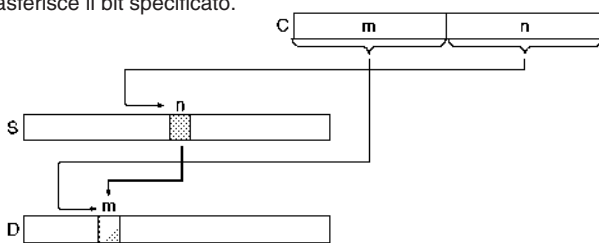
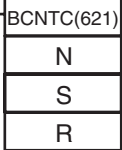
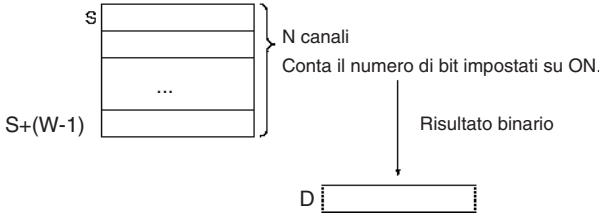
Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzioni	Posizione Condizione di esecuzione
BLOCK PROGRAM RESTART BPRS 812	BPRS (812) N **N:** numero del programma a blocchi	Mette in pausa e riavvia il programma a blocchi specificato da un altro programma a blocchi. BPRG a BPRS n a BEND Istruzione BPRS(812) eseguita dal programma a blocchi n. BPRG a n a BEND Programma a blocchi n. Questo programma non verrà eseguito finché il bit a rimane impostato su ON.	Programma a blocchi Richiesta
CONDITIONAL BLOCK EXIT EXIT 806	EXIT(806) **B**: bit operando	EXIT(806) senza un bit operando esce dal programma se la condizione di esecuzione è ON. BPRG Condizione di esecuzione OFF Condizione di esecuzione ON A A eseguito A eseguito Condizione di esecuzione EXIT B B eseguito BEND Raggiunta fine del blocco	Programma a blocchi Richiesta
CONDITIONAL BLOCK EXIT EXIT 806	EXIT(806)B **B**: bit operando	EXIT(806) senza un bit operando esce dal programma se la condizione di esecuzione è ON. BPRG Bit operando OFF (ON per EXIT NOT) Bit operando ON (OFF per EXIT NOT) A A eseguito A eseguito EXIT B (EXIT NOT B) B B eseguito BEND Raggiunta fine del blocco	Programma a blocchi Richiesta
CONDITIONAL BLOCK EXIT NOT EXIT NOT 806	EXIT NOT(806) B **B**: bit operando	EXIT(806) senza un bit operando esce dal programma se la condizione di esecuzione è OFF.	Programma a blocchi Richiesta

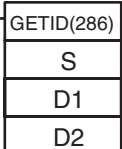
Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzioni**	**Posizione** **Condizione di esecuzione**
CONDITIONAL BLOCK BRANCHING IF 802	IF (802)	Se la condizione di esecuzione è ON, vengono eseguite le istruzioni comprese tra IF(802) ed ELSE(803), mentre se la condizione di esecuzione è OFF vengono eseguite le istruzioni tra ELSE(803) e IEND(804). Condizione di esecuzione / IF / A / ELSE / B / IEND Condizione di esecuzione ON? NO / YES A eseguito (tra IF ed ELSE) / B eseguito (dopo ELSE) / IEND	Programma a blocchi Richiesta
CONDITIONAL BLOCK BRANCHING IF 802	IF (802) B **B**: bit operando	Se il bit operando è impostato su ON, vengono eseguite le istruzioni comprese tra IF(802) ed ELSE(803). Se il bit operando è impostato su OFF, vengono eseguite le istruzioni comprese tra ELSE(803) e IEND(804). IF R (IF NOT R) / A / ELSE / B / IEND Bit operando ON? NO / YES A eseguito (tra IF ed ELSE) / B eseguito (dopo ELSE) / IEND	Programma a blocchi Richiesta
CONDITIONAL BLOCK BRANCHING (NOT) IF NOT 802	IF (802) NOT B **B**: bit operando	Le istruzioni comprese tra IF(802) ed ELSE(803) vengono eseguite se il bit operando è impostato su ON, mentre le istruzioni tra ELSE(803) e IEND(804) vengono eseguite se il bit operando è impostato su OFF.	Programma a blocchi Richiesta
CONDITIONAL BLOCK BRANCHING (ELSE) ELSE 803	---	Se l'istruzione ELSE(803) viene omessa e il bit operando è impostato su ON, verranno eseguite le istruzioni comprese tra IF(802) e IEND(804).	Programma a blocchi Richiesta
CONDITIONAL BLOCK BRANCHING END IEND 804	---	Se il bit operando è impostato su OFF, verranno eseguite solo le istruzioni inserite dopo IEND(804).	Programma a blocchi Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzioni**	**Posizione** **Condizione di esecuzione**
ONE CYCLE AND WAIT WAIT 805	WAIT(805)	Se la condizione di esecuzione è ON per WAIT(805), vengono ignorate le rimanenti istruzioni del programma a blocchi. Condizione di esecuzione OFF / Condizione di esecuzione OFF / Condizione di esecuzione ON BPRG A Condizione di esecuzione WAIT B BEND C A eseguito B eseguito C eseguito / C eseguito / C eseguito attesa	Programma a blocchi Richiesta
ONE CYCLE AND WAIT WAIT 805	WAIT(805) B **B**: bit operando	Se il bit operando è impostato su OFF (ON per WAIT(805) NOT), le istruzioni rimanenti del programma a blocchi verranno ignorate. Nel ciclo successivo non verrà eseguita alcuna istruzione del programma a blocchi, ad eccezione della condizione di esecuzione per WAIT(805) o WAIT(805) NOT. Quando la condizione di esecuzione passa a ON (OFF per WAIT(805) NOT), verranno eseguite le istruzioni da WAIT(805) o WAIT(805) NOT fino alla fine del programma.	Programma a blocchi Richiesta
ONE CYCLE AND WAIT (NOT) WAIT NOT 805	WAIT(805) NOT B **B**: bit operando	Se il bit operando è impostato su OFF (ON per WAIT(805) NOT), le istruzioni rimanenti del programma a blocchi verranno ignorate. Nel ciclo successivo non verrà eseguita alcuna istruzione del programma a blocchi, ad eccezione della condizione di esecuzione per WAIT(805) o WAIT(805) NOT. Quando la condizione di esecuzione passa a ON (OFF per WAIT(805) NOT), verranno eseguite le istruzioni da WAIT(805) o WAIT(805) NOT fino alla fine del programma.	Programma a blocchi Richiesta
TIMER WAIT TIMW 813 (BCD) TIMWX 816 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	TIMW(813) N SV **N**: numero di temporizzatore **SV**: Valore impostato TIMWX(816) N SV **N**: numero di temporizzatore **SV**: Valore impostato	Ritarda l'esecuzione del programma a blocchi del periodo di tempo specificato. L'esecuzione riprende dall'istruzione dopo TIMW(813)/TIMWX(816) al completamento del conteggio SV: Da 0 a 999,9 secondi per il formato BCD e da 0 a 6.553,5 secondi per il formato binario BPRG A TIMW N S B BEND C A eseguito SV man-tenuto Tempo trascorso B eseguito C eseguito	Programma a blocchi Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzioni	Posizione Condizione di esecuzione
COUNTER WAIT CNTW 814 (BCD) CNTWX 817 (binario) (solo Moduli CS1-H, CJ1-H, CJ1M o CS1D)	CNTW(814) N SV **N**: numero di contatore **SV**: Valore impostato **I**: ingresso di conteggio CNTWX(817) N SV **N**: numero di contatore **SV**: Valore impostato **I**: ingresso di conteggio	Ritarda l'esecuzione delle rimanenti istruzioni del programma a blocchi fino al completamento del conteggio specificato. L'esecuzione riprende dall'istruzione dopo CNTW(814)/CNTWX(817) al completamento del conteggio. SV: Da 0 a 9.999 volte per il formato BCD e da 0 a 65.535 volte per il formato binario BPRG A TIMW N S R B BEND C A eseguito SV man-tenuto Tempo trascorso B eseguito C eseguito C eseguito C eseguito	Programma a blocchi Richiesta
HIGH-SPEED TIMER WAIT TMHW 815 (BCD) TMHWX 818 (binario) (solo CS1-H, CJ1-H, CJ1M o CS1D)	TMHW(815) N SV **N**: numero di temporizzatore **SV**: Valore impostato TMHW(818) N SV **N**: numero di temporizzatore **SV**: Valore impostato	Ritarda l'esecuzione delle rimanenti istruzioni del programma a blocchi del periodo di tempo specificato. L'esecuzione riprende dall'istruzione dopo TMHW(815) allo scadere del temporizzatore. SV: Da 0 a 99,99 secondi per il formato BCD e da 0 a 655,35 secondi per il formato binario BPRG A TMHW N S B BEND C A eseguito SV im-pos-tato Tempo trascorso B eseguito C eseguito	Programma a blocchi Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzioni**	**Posizione** **Condizione di esecuzione**
LOOP LOOP 809	---	LOOP(809) specifica l'inizio del programma ciclico. Condizione di esecu- zione ON · Condizione di esecu- zione OFF · Condizione di esecu- zione OFF · Condizione di esecu- zione OFF BPRG A LOOP B Condizione di esecuzione LEND C BEND Ciclo ripetuto	Programma a blocchi Richiesta
LEND LEND 810	LEND(810)	Le istruzioni LEND(810) e LEND(810) NOT specificano la fine del ciclo. Quando viene raggiunta l'istruzione LEND(810) o LEND(810) NOT, l'esecuzione del programma torna ciclicamente all'istruzione LOOP(809) immediatamente precedente finché il bit operando per LEND(810) o LEND(810) NOT non passa, rispettivamente, a ON o a OFF oppure finché lo stato della condizione di esecuzione per LEND(810) è ON.	Programma a blocchi Richiesta
LEND LEND 810	LEND(810) B **B**: bit operando	Se il bit operando è OFF per LEND(810) (o ON per LEND(810) NOT), l'esecuzione del ciclo viene ripetuta a partire dall'istruzione dopo LOOP(809). Se il bit operando è ON per LEND(810) (o OFF per LEND(810) NOT), il ciclo viene interrotto e l'esecuzione continua a partire dall'istruzione dopo LEND(810) o LEND(810) NOT. Bit operando ON · Bit operando OFF · Bit operando OFF · Bit operando OFF BPRG A LOOP B LEND R (LEND NOT R) C BEND Ciclo ripetuto **Nota** Lo stato del bit operando risulta invertito per LEND(810) NOT.	Programma a blocchi Richiesta
LEND NOT LEND NOT 810	LEND(810) NOT **B**: bit operando	Le istruzioni LEND(810) e LEND(810) NOT specificano la fine del ciclo. Quando viene raggiunta l'istruzione LEND(810) o LEND(810) NOT, l'esecuzione del programma torna ciclicamente all'istruzione LOOP(809) immediatamente precedente finché il bit operando per LEND(810) o LEND(810) NOT non passa, rispettivamente, a ON o a OFF oppure finché lo stato della condizione di esecuzione per LEND(810) è ON.	Programma a blocchi Richiesta

3-31 Istruzioni di elaborazione delle stringhe di testo

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
MOV STRING MOV$ @MOV$ 664	MOV$(664) S D **S:** primo canale sorgente **D:** primo canale di destinazione	Trasferisce una stringa di testo.	Uscita Richiesta
CONCATENATE STRING +$ @+$ 656	+$(656) S1 S2 D **S1:** stringa di testo 1 **S2:** stringa di testo 2 **D:** Primo canale di destinazione	Concatena due stringhe di testo.	Uscita Richiesta
GET STRING LEFT LEFT$ @LEFT$ 652	LEFT$(652) S1 S2 D **S1:** primo canale stringa di testo **S2:** numero di caratteri **D:** Primo canale di destinazione	Legge il numero di caratteri specificato a partire dall'inizio della stringa di testo.	Uscita Richiesta
GET STRING RIGHT RGHT$ @RGHT$ 653	RGHT$(653) S1 S2 D **S1:** primo canale stringa di testo **S2:** numero di caratteri **D:** Primo canale di destinazione	Legge il numero di caratteri specificato a partire dalla fine della stringa di testo.	Uscita Richiesta
GET STRING MIDDLE MID$ @MID$ 654	MID$(654) S1 S2 S3 D **S1:** primo canale stringa di testo **S2:** numero di caratteri **S3:** posizione iniziale **D:** Primo canale di destinazione	Legge il numero di caratteri specificato a partire da una determinata posizione all'interno della stringa di testo.	Uscita Richiesta

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
FIND IN STRING FIND\$ @FIND\$ 660	FIND\$(660) S1 S2 D **S1:** primo canale stringa di testo sorgente **S2:** primo canale stringa di testo di ricerca **D:** Primo canale di destinazione	Cerca la stringa di testo specificata all'interno di un'altra stringa. Dati trovati S1→ A B / C D / E F / NUL NUL; S2→ C NUL; D→ 00 03	Uscita Richiesta
STRING LENGTH LEN\$ @LEN\$ 650	LEN\$(650) S D **S:** primo canale stringa di testo **D:** primo canale di destinazione	Calcola la lunghezza di una stringa di testo. S1→ 1 2 / 3 4 / 5 NUL; D 00 05	Uscita Richiesta
REPLACE IN STRING RPLC\$ @RPLC\$ 661	RPLC\$(654) S1 S2 S3 S4 D **S1:** primo canale stringa di testo **S2:** primo canale stringa di testo sostitutiva **S3:** numero di caratteri **S4:** posizione iniziale **D:** Primo canale di destinazione	Sostituisce una stringa di testo con quella specificata a partire da una determinata posizione. S3 00 04 S1→ A B / C D / E F / G H / I NUL; S2 K NUL; S4 00 05 D→ A B / C D / K G / H I / NUL NUL	Uscita Richiesta
DELETE STRING DEL\$ @DEL\$ 658	DEL\$(658) S1 S2 S3 D **S1:** primo canale stringa di testo **S2:** numero di caratteri **S3:** posizione iniziale **D:** Primo canale di destinazione	Elimina la stringa di testo specificata dall'interno di un'altra stringa. Numero di caratteri da eliminare (specificato da S2) S1→ A B / C D / E F / G H / I NUL; S3 00 05 D→ A B / C D / H I / NUL NUL	Uscita Richiesta

Istruzione **Codice mnemonico** **Codice**	**Simbolo/ operando**	**Funzione**	**Posizione** **Condizione di esecuzione**
EXCHANGE STRING XCHG$ @XCHG$ 665	XCHG$(665) Ex1 Ex2 **Ex1:** primo canale di scambio 1 **Ex2:** primo canale di scambio 2	Sostituisce la stringa di testo specificata con un'altra stringa specificata.	Uscita Richiesta
CLEAR STRING CLR$ @CLR$ 666	CLR$(666) S **S:** primo canale stringa di testo	Sostituisce il contenuto di una stringa di testo con NUL (0 esadecimale).	Uscita Richiesta
INSERT INTO STRING INS$ @INS$ 657	INS$(657) S1 S2 S3 D **S1:** primo canale stringa di testo di base **S2:** primo canale stringa di testo immessa **S3:** posizione iniziale **D:** Primo canale di destinazione	Inserisce la stringa di testo specificata all'interno di un'altra stringa. Caratteri immessi	Uscita Richiesta
Confronto di stringhe LD, AND, OR + =$, <>$, <$, <=$, >$, >=$ 670 (=$) 671 (<>$) 672 (<$) 673 (<=$) 674 (>$) 675 (>=$)	LD Simbolo S1 S2 AND Simbolo S1 S2 OR Simbolo S1 S2 **S1:** stringa di testo 1 **S2:** stringa di testo 2	Le istruzioni di confronto delle stringhe (=$, <>$, <$, <=$, >$, >=$) confrontano due stringhe di testo a partire dall'inizio, verificando i valori dei codici ASCII. Se il risultato del confronto è vero, viene creata una condizione di esecuzione ON per LOAD, AND o OR.	LD: non richiesta AND, OR: richiesta

3-32 Istruzioni di controllo dei task

Istruzione Codice mnemonico Codice	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
TASK ON TKON @TKON 820	TKON(820) N **N:** numero di task	Rende il task specificato eseguibile. Il numero di task del task specificato è superiore a quello del task locale (m<n). Il numero di task del task specificato è inferiore a quello del task locale (m>n).	Uscita Richiesta
TASK OFF TKOF @TKOF 821	TKOF(821) N **N:** numero di task	Mette in stato di attesa il task specificato. Il numero di task del task specificato è superiore a quello del task locale (m<n). Il numero di task del task specificato è inferiore a quello del task locale (m>n).	Uscita Richiesta

3-33 Istruzioni di conversione del modello (solo CPU versione 3.0 o successiva)

Istruzione Codice mnemonico	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
BLOCK TRANSFER XFERC @XFERC 565	XFERC(565) N S D **N**: numero di canali **S**: primo canale sorgente **D**: primo canale didestinazione	Trasferisce il numero di canali consecutivi specificato. S ... S+(W–1) W canali → D ... D+(W–1)	Uscita Richiesta
SINGLE WORD DISTRIBUTE DISTC @DISTC 566	DISTC(566) S Bs Of **S**: canale sorgente **Bs**: indirizzo base di destinazione **Of**: offset	Trasferisce il canale sorgente in un canale di destinazione calcolato aggiungendo il valore di offset all'indirizzo di base. S1 D S2 n n D+n È anche possibile scrivere in uno stack (operazione PUSH ONTO STACK).	Uscita Richiesta
DATA COLLECT COLLC @COLLC 567	COLLC(567) Bs Of D **Bs**: indirizzo base sorgente **Of**: Offset (BCD) **D**: canale di destinazione	Trasferisce il canale sorgente, calcolato aggiungendo un valore di offset all'indirizzo di base, nel canale di destinazione. S1 S2 n n S1+n D È anche possibile leggere da uno stack (operazione STACK DATA READ).	Uscita Richiesta

Istruzione Codice mnemonico	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
MOVE BIT MOVBC @MOVBC 568	MOVBC(568) S C D **S**: dati o canale sorgente **C**: canale di controllo (BCD) **D**: canale di destinazione	Trasferisce il bit specificato. C m n S n D m	Uscita Richiesta
BIT COUNTER BCNTC @BCNTC 621	BCNTC(621) N S R **N**: numero di canali (BCD) **S**: primo canale sorgente **R**: canale del risultato	Conta il numero totale di bit impostati su ON presenti nei canali specificati. S ... S+(W-1) N canali Conta il numero di bit impostati su ON. Risultato binario D	Uscita Richiesta

3-34 Istruzioni speciali per blocchi funzione:

Istruzione Codice mnemonico	Simbolo/ operando	Funzione	Posizione Condizione di esecuzione
GET VARIABLE ID GETID @GETID 286	GETID(286) S D1 D2 **S**: variabile o indirizzo **D1**: codice ID **D2**: canale di destinazione	Invia il codice del tipo di variabile del comando FINS (area dati) e l'indirizzo di canale per la variabile o l'indirizzo specificato. Questa istruzione viene in genere utilizzata per ottenere l'indirizzo assegnato a una variabile in un blocco funzione.	Uscita Richiesta

CAPITOLO 4
Task

In questo capitolo viene descritto il funzionamento dei task.

4-1 Caratteristiche dei task

4-1-1 Informazioni generali

Le operazioni di controllo dei PLC delle serie CS e CJ possono essere suddivise in base a funzioni, dispositivi controllati, processi, interventi da parte di sviluppatori o qualsiasi altro criterio ed è possibile programmare ciascuna operazione in un'unità distinta definita "task". L'utilizzo dei task assicura i seguenti vantaggi:

1,2,3...

1. Possibilità di affidare lo sviluppo dei programmi a più programmatori contemporaneamente.

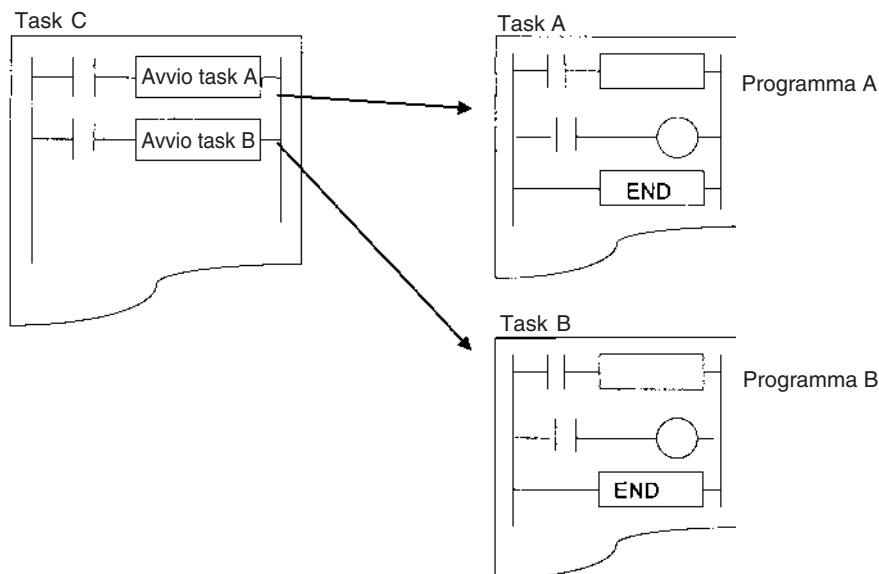
   È infatti possibile assemblare parti di programma progettate singolarmente in un unico programma utente con il minimo sforzo.
2. Possibilità di standardizzare i programmi in moduli.

   Più specificamente, è possibile combinare le funzioni dei dispositivi di programmazione indicate di seguito per sviluppare programmi che andranno a costituire moduli standard autonomi anziché programmi progettati per determinati sistemi, ovvero macchine o dispositivi specifici. Ciò significa che è possibile combinare facilmente programmi sviluppati separatamente da più persone.
   - Programmazione mediante simboli
   - Designazione globale e locale di simboli
   - Assegnazione automatica di simboli locali a indirizzi
3. Miglioramento della risposta complessiva.

   La risposta complessiva risulta migliorata perché il sistema viene suddiviso in un programma di controllo globale e in singoli programmi di controllo, offrendo la possibilità di eseguire solo i programmi specifici in base alle esigenze.
4. Operazioni di modifica e debug semplificate.
   - Il debug è molto più efficiente dal momento che i task possono essere sviluppati separatamente da più programmatori e, di conseguenza, le successive operazioni di modifica e debug vengono eseguite sui singoli task.
   - La manutenzione risulta semplificata, in quanto verrà modificato solo il task che necessita di revisione al fine di creare nuove specifiche o apportare altre modifiche.
   - Il debug è più efficiente poiché risulta semplice determinare se un indirizzo è specifico o globale. Inoltre, durante il debug sarà necessario controllare gli indirizzi tra i programmi una sola volta, dal momento che i simboli vengono designati a livello globale o locale e i simboli locali vengono assegnati automaticamente agli indirizzi tramite i dispositivi di programmazione.
5. Passaggio da un programma all'altro semplificato.

   Quando è necessario modificare il funzionamento, è possibile inserire nel programma un'istruzione di controllo dei task che consente di eseguire task specifici per determinati prodotti (programmi).

6. Agevolazione della comprensione dei programmi utente.

   I programmi sono strutturati in blocchi che semplificano notevolmente la comprensione delle sezioni di programma che verrebbero convenzionalmente gestite mediante istruzioni di salto.

4-1-2 Task e programmi

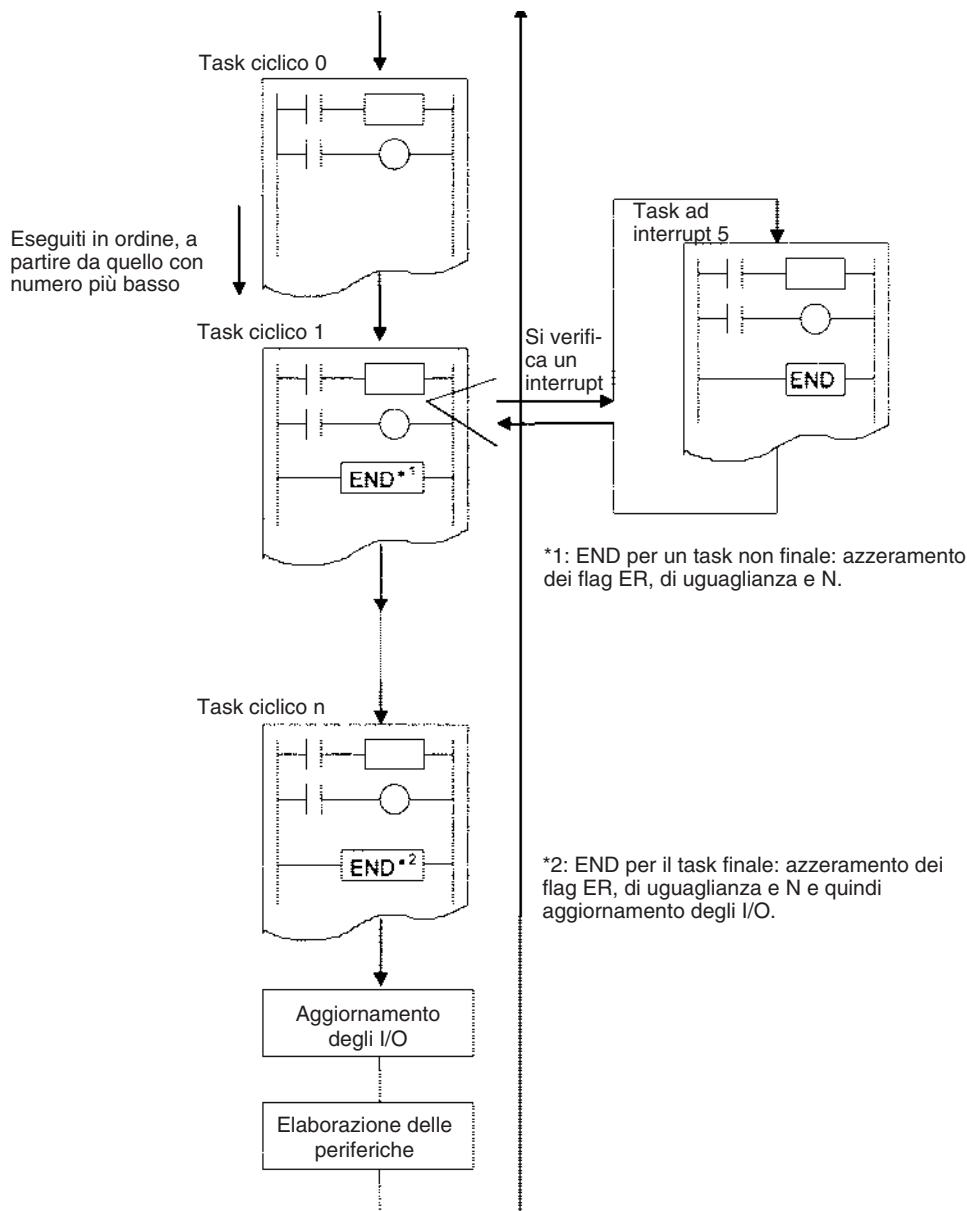
- È possibile controllare fino a 288 programmi (task). I singoli programmi vengono assegnati ai task in un rapporto di 1:1. I task possono essere suddivisi nelle seguenti categorie di massima:
- Task ciclici
- Task ad interrupt

Nota

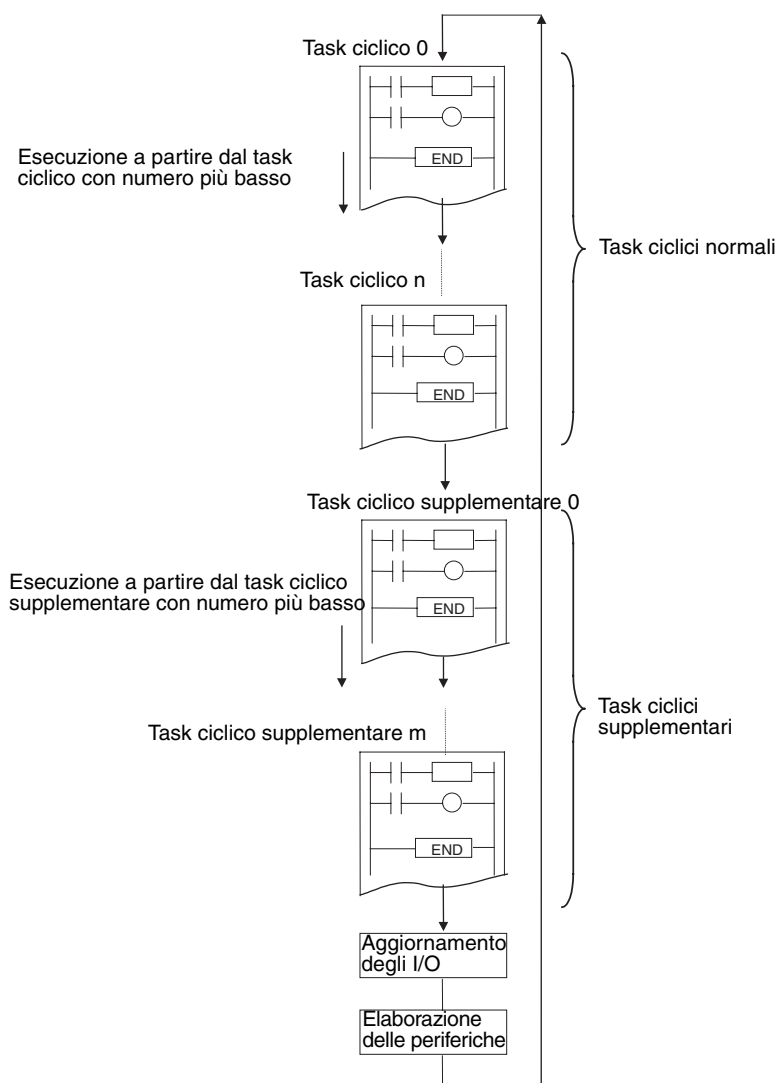
1. È possibile creare fino a 32 task ciclici e 256 task ad interrupt per un totale di 288 task. A ciascun task viene assegnato un numero univoco compreso tra 0 e 31 per i task ciclici e 0 e 255 per i task ad interrupt.
2. Nel caso delle CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt (numeri di task da 0 a 255) come task ciclici avviandoli tramite l'istruzione TKON. Tali task vengono definiti "task ciclici supplementari". Se si utilizzano task ciclici supplementari, il numero totale di task ciclici che è possibile utilizzare sarà 288.
3. Le CPU CJ1 attualmente non supportano task ad interrupt esterni e task ad interrupt di I/O. Il numero massimo di task per una CPU CJ1 è quindi 35, ovvero 32 task ciclici e 3 task ad interrupt. Pertanto, anche il numero massimo di programmi che è possibile creare e gestire è 35.

È necessario che ciascun programma assegnato a un task termini con un'istruzione END(001). L'aggiornamento degli I/O avrà luogo solo dopo l'esecuzione di tutti i programmi di task all'interno di un ciclo.

4-1-3 Funzionamento di base della CPU

La CPU esegue i task ciclici, inclusi i task ciclici supplementari (solo CPU CS1-H, CJ1-H, CJ1M o CS1D), a partire dal numero più basso. Inoltre, se viene ricevuto un interrupt, la CPU interrompe l'esecuzione dei task ciclici per consentire l'esecuzione del task ad interrupt.

Nota Tutti i flag di condizione (ER, CY, EQ, AER e così via) e le condizioni di istruzione (attivazione dell'interblocco e così via) vengono cancellati all'avvio di un task. Non è pertanto possibile leggere i flag di condizione né ripartire le istruzioni INTERLOCK/INTERLOCK CLEAR (IL/ILC), JUMP/JUMP END (JMP/JME) o SUBROUTINE CALL/SUBROUTINE ENTRY (SBS/SBN) tra due task.

Nel caso delle CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt come task ciclici avviandoli tramite l'istruzione TKON. Tali task vengono definiti "task ciclici supplementari". I task ciclici supplementari (numeri di task da 0 a 255) vengono eseguiti a partire dal numero di task più

basso dopo il completamento dell'esecuzione del task ciclico normale (numeri di task da 0 a 31).

4-1-4 Tipi di task

I task vengono sommariamente classificati come task ciclici o task ad interrupt. I task ad interrupt sono ulteriormente suddivisi in task ad interrupt di spegnimento, task ad interrupt programmati, task ad interrupt di I/O (solo serie CS) e task ad interrupt esterni (solo serie CS). È inoltre possibile eseguire i task ad interrupt come task ciclici supplementari.

Nota Nel caso delle CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt come task ciclici avviandoli tramite l'istruzione TKON. Tali task vengono definiti "task ciclici supplementari".

Task ciclici

I task ciclici con stato READY vengono eseguiti una volta ogni ciclo, dall'inizio del programma fino all'istruzione END(001), in ordine numerico a partire dal task con numero più basso. Il numero massimo di task ciclici è 32 (numeri di task da 00 a 31).

Nota Nel caso delle CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt (numeri di task da 0 a 255) in modo ciclico analogamente ai task ciclici normali (numeri di task da 0 a 31). Se si utilizzano task ciclici supplementari, il numero totale di task ciclici che è possibile utilizzare sarà 288.

Task ad interrupt

Se si verifica una condizione di interrupt, viene eseguito un task ad interrupt anche se è in corso l'esecuzione di un task ciclico o di un task ciclico supplementare. Il task ad interrupt viene eseguito in qualsiasi momento del ciclo, anche durante l'esecuzione del programma utente, l'aggiornamento degli I/O o la gestione delle periferiche, purché venga soddisfatta la condizione di esecuzione per l'interrupt.

Nel caso delle CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile eseguire i task ad interrupt come task ciclici. Le CPU CS1D per sistemi a due CPU non supportano gli interrupt. Nel caso delle CPU CS1D per sistemi a due CPU, è possibile utilizzare i task ad interrupt solo come task ciclici supplementari.

Per attivare i task ad interrupt, è possibile utilizzare gli ingressi di interrupt integrati e gli ingressi del contatore ad alta velocità di una CPU CJ1M. Per ulteriori informazioni, fare riferimento al *Manuale dell'operatore degli I/O integrati della serie CJ*.

Task ad interrupt di spegnimento

Il task ad interrupt di spegnimento viene eseguito quando l'alimentazione alla CPU viene interrotta. È possibile programmare un solo task ad interrupt di spegnimento (numero di task 1).

Nota È necessario che il task ad interrupt di spegnimento venga eseguito prima della scadenza del tempo indicato di seguito. In caso contrario, il task verrà chiuso in modo forzato.

10 ms - ritardo di rilevamento della caduta di tensione

Il ritardo di rilevamento della caduta di tensione viene impostato nella configurazione del PLC.

Task ad interrupt programmati

Un task ad interrupt programmato viene eseguito a intervalli regolari in base al temporizzatore interno della CPU. Il numero massimo di task ad interrupt programmati è 2 (numeri di task 2 e 3).

Nota Per impostare l'interrupt per un task ad interrupt programmato, utilizzare l'istruzione SET INTERRUPT MASK (MSKS(690)). È possibile impostare i tempi di interrupt con incrementi di 10 ms o 1,0 ms nella configurazione del PLC.

Task ad interrupt di I/O

Un task ad interrupt di I/O viene eseguito nel caso in cui venga attivato un ingresso del Modulo di interrupt di ingresso. Il numero massimo di task ad interrupt di I/O è 32 (numeri di task da 100 a 131). È necessario che il Modulo di interrupt di ingresso sia installato nel sistema CPU. Nel caso delle CPU CJ1-H è necessario collegare tale Modulo in prossimità della CPU, in uno dei cinque slot (da 0 a 4) disponibili. Nel caso delle CPU CJ1M è necessario collegare tale Modulo in prossimità della CPU, in uno dei tre slot (da 0 a 2) disponibili. Per richiedere l'esecuzione dei task ad interrupt di I/O, non è possibile utilizzare Moduli di interrupt di I/O installati in posizioni diverse da quelle indicate.

Le CPU CJ1 non supportano gli interrupt di I/O.

Task ad interrupt esterni

Un task ad interrupt esterno viene eseguito quando richiesto dal programma utente di un Modulo di I/O speciale, di un Modulo CPU bus o di una scheda interna (solo serie CS). È tuttavia necessario che i Moduli di I/O speciale e i Moduli CPU bus siano installati nel sistema CPU. Il Modulo di I/O speciale o il Modulo CPU bus deve essere installato nel sistema CPU. Nel caso delle CPU CJ1-H è necessario collegare tale Modulo in prossimità della CPU, in uno dei cinque slot (da 0 a 4) disponibili. Nel caso delle CPU CJ1M è necessario collegare tale Modulo in prossimità della CPU, in uno dei tre slot (da 0 a 2) disponibili. Per generare interrupt esterni, non è possibile utilizzare Moduli installati in posizioni diverse da quelle indicate.

Il numero massimo di task ad interrupt esterni è 256 (numeri di task da 0 a 255). Se il numero di un task ad interrupt esterno coincide con quello di un task ad interrupt di spegnimento, un task ad interrupt programmato o un task ad interrupt di I/O, verrà eseguito il task associato a una delle due condizioni di interrupt (le condizioni sono regolate da una logica di tipo OR) ma, in genere, è consigliabile non assegnare numeri di task duplicati.

Le CPU CJ1 non supportano gli interrupt di I/O.

Task ciclici supplementari (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)

I task ad interrupt possono essere eseguiti ad ogni ciclo analogamente ai task ciclici normali. I task ciclici supplementari (numeri di task da 0 a 255) vengono eseguiti a partire dal numero di task più basso dopo il completamento dell'esecuzione del task ciclico normale (numeri di task da 0 a 31). Il numero massimo di task ciclici supplementari è 256 (numeri di task da 0 a 255). I task ad interrupt eseguiti ciclicamente, tuttavia, si differenziano dai task ciclici normali per il fatto che vengono avviati tramite l'istruzione TKON(820). Inoltre, non è possibile utilizzare le istruzioni TKON(820) e TKOF all'interno di task ciclici supplementari, vale a dire non è possibile controllare task ciclici normali né altri task ciclici supplementari da un task ciclico supplementare.

Se il numero di un task ciclico supplementare coincide con quello di un task ad interrupt di spegnimento, un task ad interrupt programmato o un task ad interrupt di I/O, verrà eseguito il task associato a una delle due condizioni di interrupt (le condizioni sono regolate da una logica di tipo OR). Non utilizzare i task ad interrupt sia come task ad interrupt normali che come task ciclici supplementari.

Nota

1. Il task ad interrupt di spegnimento, descritto al punto 1) di cui sopra, ha priorità e viene eseguito quando l'alimentazione viene interrotta anche se è in esecuzione un altro task ad interrupt.
2. Se, quando viene inviato un interrupt programmato, un interrupt di I/O o un interrupt esterno, è in esecuzione un altro task ad interrupt, tali task non verranno eseguiti finché il task ad interrupt correntemente in esecuzione non viene completato. Se vengono inviati contemporaneamente più interrupt, i task ad interrupt verranno eseguiti in sequenza, a partire dal task ad interrupt con numero più basso.

3. Le differenze tra task ciclici normali e task ciclici supplementari sono illustrate nella tabella riportata di seguito.

Elemento	Task ciclici supplementari	Task ciclici normali
Attivazione all'avvio	Impostazione non prevista	Impostazione tramite CX-Programmer
Utilizzo delle istruzioni TKON/TKOF	Non consentito	Consentito
flag di task	Non supportato	Supportati (i numeri di task ciclici da 00 a 31 corrispondono ai flag di task da TK00 a TK31).
Flag di esecuzione del task iniziale (A20015) e flag di inizio task (A20014)	Non supportato	Supportati
Valori dei registri indice (IR) e dati (DR)	Non definiti all'avvio del task (come per i task ad interrupt normali). All'inizio di ciascun ciclo i valori risultano non definiti. Impostare sempre i valori prima di utilizzarli. Non è possibile leggere i valori impostati nel ciclo precedente.	Non definiti all'avvio del funzionamento. È possibile leggere i valori impostati nel ciclo precedente.

4. Le CPU CJ1 non supportano task ad interrupt di I/O e task ad interrupt esterni.

4-1-5 Impostazioni e condizioni di esecuzione dei task

Nella tabella riportata di seguito viene fornita una descrizione delle condizioni di esecuzione dei task, delle impostazioni correlate e dello stato dei task.

Task		Numero	Condizione di esecuzione	Impostazione correlata
Task ciclici		da 0 a 31	Se lo stato è READY, qualora impostati per l'attivazione all'avvio o mediante istruzione TKON(820), vengono eseguiti una volta ad ogni ciclo quando ottengono l'autorizzazione all'esecuzione.	Nessuno
Task ad interrupt	Task ad interrupt di spegnimento	Task ad interrupt 1	Viene eseguito allo spegnimento della CPU.	• Il task ad interrupt di spegnimento viene abilitato nella configurazione del PLC.
	Task ad interrupt programmati 0 e 1	Task ad interrupt 2 e 3	Vengono eseguiti una volta quando scade l'intervallo di tempo preimpostato in base al temporizzatore interno della CPU.	• Il tempo del task ad interrupt programmato (da 0 a 9999) viene impostato tramite l'istruzione SET INTERRUPT MASK (MSKS). • L'unità di tempo per i task programmati (10 ms o 1,0 ms) viene impostata nella configurazione del PLC.
	Task ad interrupt di I/O da 00 a 31	Task ad interrupt da 100 a 131	Vengono eseguiti quando viene attivato un ingresso di un Modulo di interrupt di ingresso installato nel sistema CPU.	• Il mascheramento per gli ingressi specificati viene annullato tramite l'istruzione SET INTERRUPT MASK (MSKS).
	Task ad interrupt esterni da 0 a 255	Task ad interrupt da 0 a 255	Vengono eseguiti quando richiesto dal programma utente di un Modulo di I/O speciale o un Modulo CPU bus installato nel sistema CPU o dal programma utente di una scheda interna (solo serie CS).	Nessuna (sempre abilitati)
Task ciclici supplementari (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)		Task ad interrupt da 0 a 255	Se lo stato è READY, qualora impostati per l'avvio mediante istruzione TKON(820), vengono eseguiti una volta ad ogni ciclo quando ottengono l'autorizzazione all'esecuzione.	Nessuna (sempre abilitati)

Nota

1. È necessario che il Modulo di interrupt di ingresso sia installato nel sistema CPU. Nel caso delle CPU CJ1-H è necessario collegare tale Modulo in prossimità della CPU, in uno dei cinque slot (da 0 a 4) disponibili. Nel caso delle CPU CJ1M è necessario collegare tale Modulo in prossimità della CPU, in uno dei tre slot (da 0 a 2) disponibili. Per richiedere l'esecuzione dei task ad interrupt di I/O, non è possibile utilizzare Moduli di interrupt di I/O installati in posizioni diverse da quelle indicate.
2. Il Modulo di I/O speciale o il Modulo CPU bus deve essere installato nel sistema CPU. Nel caso delle CPU CJ1-H è necessario collegare tale Modulo in prossimità della CPU, in uno dei cinque slot (da 0 a 4) disponibili. Nel caso delle CPU CJ1M è necessario collegare tale Modulo in prossimità della CPU, in uno dei tre slot (da 0 a 2) disponibili. Per generare interrupt esterni, non è possibile utilizzare Moduli installati in posizioni diverse da quelle indicate.
3. Quando l'operazione di cancellazione della memoria viene eseguita da una Console di programmazione, il numero di task ciclici e di task ad interrupt che è possibile creare è limitato.

- È possibile creare solo il task ciclico 0.
  Non è possibile creare i task ciclici da 1 a 31 con una Console di programmazione. È tuttavia possibile modificare tali task nel caso in cui siano già stati creati con CX-Programmer.
- È possibile creare solo i task ad interrupt 1, 2, 3 e da 100 a 131 (solo serie CS).
  Non è possibile creare i task ad interrupt 0, da 4 a 99 e da 132 a 255 con una Console di programmazione (a eccezione dei task da 140 a 143, che possono essere creati per le CPU CJ1M).
  È tuttavia possibile modificare tali task nel caso in cui siano già stati creati con CX-Programmer.

4-1-6 Stato dei task ciclici

In questa sezione viene descritto lo stato dei task ciclici, inclusi i task ciclici supplementari supportati solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

I task ciclici hanno sempre uno dei seguenti quattro stati: Disabilitato, READY (pronto), RUN (eseguibile) e WAIT (in attesa).

Stato Disabilitato (INI)

Un task con stato Disabilitato non viene eseguito. Lo stato di tutti i task ciclici viene impostato su Disabilitato in modalità PROGRAM. Ogni volta che un task ciclico passa da tale stato a un altro, è possibile ripristinare lo stato Disabilitato solo tornando nuovamente alla modalità PROGRAM.

Stato READY

Per controllare il momento in cui lo stato del task passa a READY, è possibile impostare un attributo del task. L'attributo può essere impostato per l'attivazione del task tramite l'istruzione TASK ON o all'avvio della modalità operativa RUN.

Task attivati da istruzioni

Per passare dallo stato Disabilitato, o dallo stato di attesa, allo stato READY di un task ciclico attivato da istruzioni, è possibile utilizzare l'istruzione TASK ON (TKON(820)).

Task attivati all'avvio del funzionamento

Un task ciclico attivato all'avvio del funzionamento passa dallo stato Disabilitato allo stato READY quando la modalità operativa passa da PROGRAM a RUN o MONITOR. Questo tipo di attivazione è applicabile solo ai task ciclici normali.

Nota Per impostare uno o più task, con numero di task da 0 a 31, per il passaggio allo stato READY all'avvio del funzionamento, è possibile utilizzare un dispositivo di programmazione. Questa impostazione, tuttavia, non è prevista per i task ciclici supplementari.

Stato RUN

Quando un task ciclico ottiene l'autorizzazione all'esecuzione, il relativo stato passa da READY a RUN e il task viene eseguito.

Stato di attesa (WAIT)

Per passare dallo stato Disabilitato allo stato di attesa di un task ciclico, è possibile utilizzare un'istruzione TASK OFF (TKOF(821)).

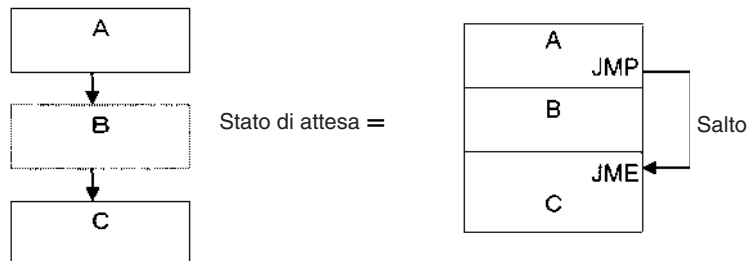
Nota Con CX-Programmer versione 4.0 o successiva è possibile monitorare in linea i programmi di task per PLC della serie CS/CJ per verificare se sono in esecuzione o sono stati interrotti. In CX-Programmer vengono visualizzate le seguenti indicazioni di stato:

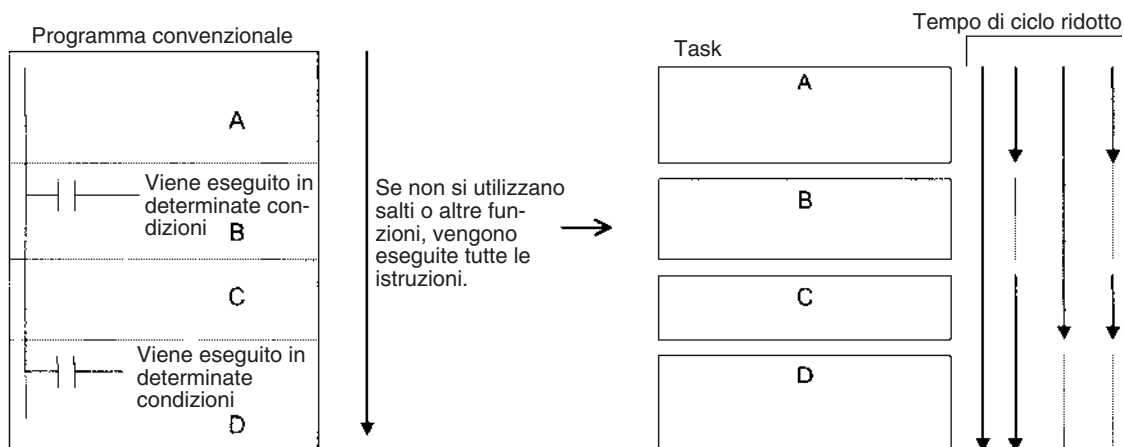
- In esecuzione: il task è nello stato READY o RUN (non è possibile differenziare tra questi due tipi di stato).
- Interrotto: il task è nello stato INI o WAIT (non è possibile differenziare tra questi due tipi di stato).

4-1-7 Passaggi di stato

Nota

1. È possibile portare in stato di attesa un task con stato RUN utilizzando l'istruzione TKOF(821) anche quando l'istruzione TKOF(821) viene eseguita all'interno di tale task.
2. L'attivazione all'avvio del funzionamento è applicabile solo ai task ciclici normali e non ai task ciclici supplementari.

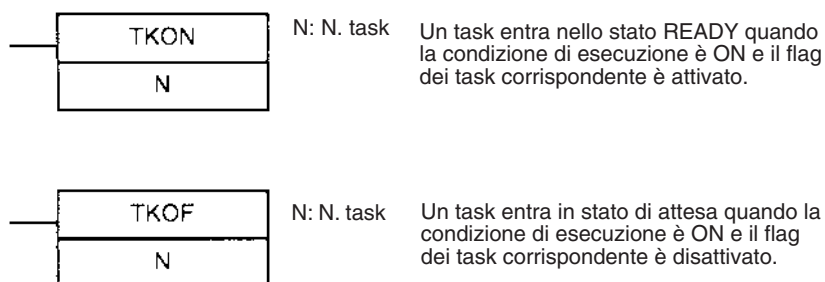
Il funzionamento dello stato di attesa è simile a quello di un salto (JMP-JME), ossia lo stato dell'uscita per il task in attesa viene mantenuto.

Poiché le istruzioni non vengono eseguite quando il task è in stato di attesa, non si avrà alcun aumento del tempo di esecuzione delle istruzioni. Per ridurre il tempo di ciclo, è possibile definire come task le sezioni di programma

che non devono necessariamente essere eseguite in modo continuo e assegnarvi lo stato di attesa.

Nota Per stato di attesa si intende semplicemente che, durante l'esecuzione dei task, un task verrà ignorato. Il passaggio allo stato di attesa non determina l'interruzione del programma.

4-2 Uso dei task

4-2-1 Istruzioni TASK ON e TASK OFF

Le istruzioni TASK ON (TKON(820)) e TASK OFF (TKOF(821)) determinano il passaggio dallo stato READY allo stato di attesa di un task ciclico, o di un task ciclico supplementare, da un programma.

Nota I task ciclici supplementari sono supportati solo dalle CPU CS1-H, CJ1-H, CJ1M e CS1D.

Per modificare in qualsiasi momento lo stato di un task ciclico da READY a WAIT (in attesa) e viceversa, è possibile utilizzare le istruzioni TASK ON e TASK OFF. Un task ciclico in stato READY o in stato di attesa manterrà tale stato nei cicli successivi.

Le istruzioni TASK ON e TASK OFF possono essere utilizzate solo con i task ciclici e non con i task ad interrupt.

Nota È necessario che in ogni ciclo sia presente almeno un task ciclico con stato READY. Se nessun task ciclico è in stato READY, verrà attivato il flag di errore di esecuzione task (A29512) e la CPU verrà arrestata.

Esempio task ciclico

A TKON 1

B TKON 2

C TKON 3

D TKOF 0

Task ciclico 0 (stato READY all'avvio del funzionamento)

Task ciclico 1

Task ciclico 2

Task ciclico 3

1) Il task 0 sarà nello stato READY all'avvio del funzionamento.

Gli altri task rimarranno nello stato Disabilitato.

Task ciclico 0 / Task ciclico 1 / Task ciclico 2 / Task ciclico 3

2) Il task 1 entrerà nello stato READY se A è ON e i task 2 e 3 rimarranno nello stato Disabilitato.

Task ciclico 0 / Task ciclico 1 / Task ciclico 2 / Task ciclico 3

3) Il task 0 entrerà nello stato di attesa se D è ON.

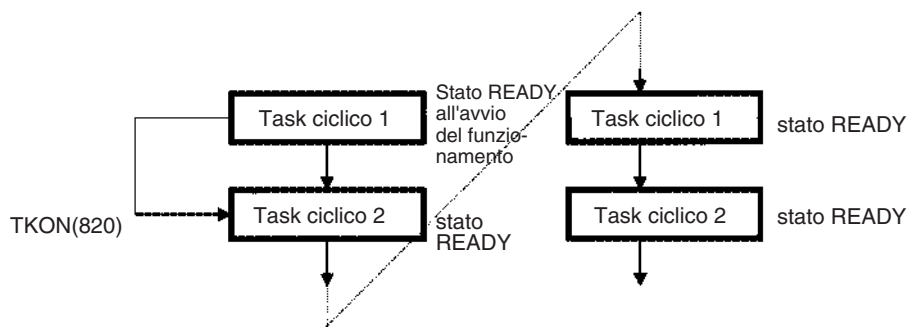
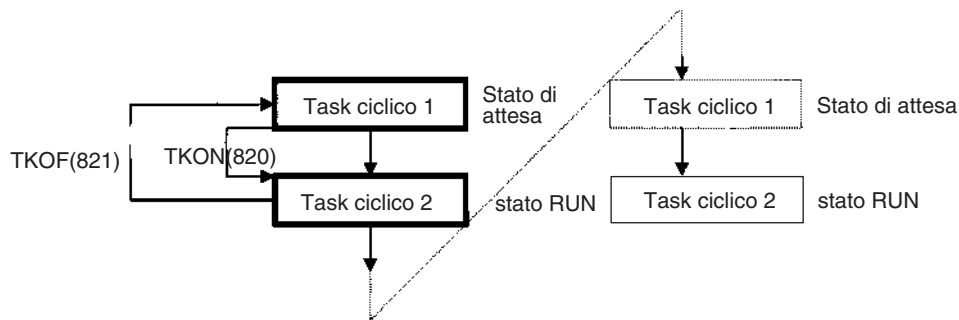
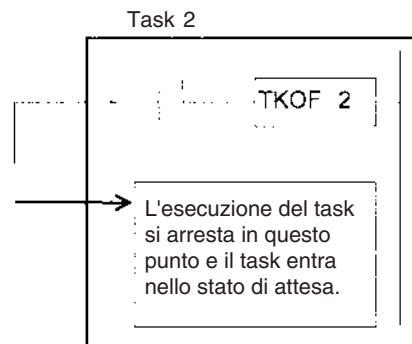
Gli altri task rimarranno nello stato in cui si trovano.

Task ciclico 0 / Task ciclico 1 / Task ciclico 2 / Task ciclico 3

stato READY

Stato di attesa/Disabilitato

Task e ciclo di esecuzione

Un task ciclico, inclusi i task ciclici supplementari, in stato READY manterrà tale stato nei cicli successivi.

Un task ciclico in stato di attesa manterrà tale stato nei cicli successivi. Per passare dallo stato di attesa allo stato READY, è necessario attivare il task mediante l'istruzione TKON(820).

Se all'interno di un task in esecuzione è presente un'istruzione TKOF(821), l'esecuzione del task si interromperà nel punto in cui viene eseguita tale istruzione e il task passerà allo stato di attesa.

Numeri di task ciclico e ciclo di esecuzione (task ciclici supplementari inclusi)

Se il task m attiva il task n e il numero del task m è superiore a quello di n, lo stato del task n passerà a READY nel ciclo successivo.

Esempio: se il task 5 attiva il task 2, lo stato del task 2 passerà a READY nel ciclo successivo.

Se il task m attiva il task n e il numero del task m è inferiore a quello di n, lo stato del task n passerà a READY nello stesso ciclo.

Esempio: se il task 2 attiva il task 5, lo stato del task 5 passerà a READY nello stesso ciclo.

Se il task m attiva lo stato di attesa per il task n e il numero del task m è superiore a quello di n, il task n passerà allo stato di attesa nel ciclo successivo.

Esempio se il task 5 attiva lo stato di attesa per il task 2, quest'ultimo passerà allo stato di attesa nel ciclo successivo.

Se il task m attiva lo stato di attesa per il task n e il numero del task m è inferiore a quello di n, il task n passerà allo stato di attesa nello stesso ciclo.

Esempio se il task 2 attiva lo stato di attesa per il task 5, quest'ultimo passerà allo stato di attesa nello stesso ciclo.

Relazione tra task e memoria I/O

Esistono due differenti modalità di utilizzo dei registri indice (IR) e dei registri dati (DR): 1) per ogni task viene utilizzato un registro indipendente o 2) i registri sono condivisi da tutti i task (funzione supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D).

Nel caso dei registri indipendenti, il registro IR0 utilizzato dal task ciclico 1 è ad esempio differente dal registro IR0 utilizzato dal task ciclico 2. Nel caso dei

registri condivisi, il registro IR0 utilizzato dal task ciclico 1 è identico al registro IR0 utilizzato dal task ciclico 2.

L'impostazione che determina se i registri sono indipendenti o condivisi viene configurata mediante CX-Programmer.

- Altri canali e bit nella memoria I/O sono condivisi da tutti i task. CIO 001000, ad esempio, è un bit condiviso sia dal task ciclico 1 che dal task ciclico 2. Pertanto, durante la programmazione, è bene prestare particolare attenzione quando si utilizzano aree della memoria I/O diverse dalle aree IR e DR, in quanto i valori modificati per un task verranno utilizzati anche da altri task.

memoria I/O	Relazione con i task
Area CIO, area ausiliaria, area di memoria dei dati e tutte le altre aree di memoria, ad eccezione di IR e DR (vedere nota 1)	Condivise con altri task.
Registri indice (IR) e registri dati (DR) (vedere nota 2)	Utilizzati separatamente per ciascun task.

Nota

1. Anche il banco EM corrente è condiviso tra i vari task. Pertanto, se ad esempio il numero del banco EM corrente viene modificato per il task ciclico 1, il nuovo numero di banco EM corrente sarà valido anche per il task ciclico 2.
2. I valori di IR e DR non vengono impostati all'avvio dei task ad interrupt (inclusi i task ciclici supplementari). Se IR e DR vengono utilizzati in un task ad interrupt, è necessario che tali valori vengano impostati mediante le istruzioni MOVR/MOVRW (MOVE TO REGISTER e MOVE TIMER/COUNTER PV TO REGISTER) all'interno del task ad interrupt. Una volta eseguito il task ad interrupt, verranno automaticamente ripristinati i valori di IR e DR precedenti all'interrupt.

Relazione tra task e funzionamento dei temporizzatori

I valori attuali di temporizzatore per le istruzioni TIM, TIMX, TIMH, TIMHX, TMHH, TMHHX, TIMW, TIMWX, TMHW e TMHWX programmate per i numeri di temporizzatore da 0000 a 2047 verranno aggiornati anche se si passa da un task a un altro o se il task contenente il temporizzatore passa allo stato di attesa o viene riportato allo stato READY.

Se il task contenente TIM passa allo stato di attesa e successivamente viene riportato allo stato READY, il flag di completamento verrà attivato se l'istruzione TIM viene eseguita quando il valore attuale è 0. I flag di completamento per i temporizzatori vengono aggiornati solo all'esecuzione dell'istruzione. Se l'istruzione TIM viene eseguita quando il valore attuale non è ancora pari a 0, quest'ultimo continuerà ad essere aggiornato così come avviene quando il task è in stato READY.

- Quando il task è in stato di attesa, i valori attuali dei temporizzatori programmati con i numeri di temporizzatore da 2048 a 4098 vengono mantenuti.

Relazione tra task e flag di condizione

Tutti i flag di condizione vengono cancellati prima dell'esecuzione di ciascun task. Pertanto, non è possibile leggere lo stato dei flag di condizione al termine del task 1 dal task 2. Tuttavia, con le CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile utilizzare le istruzioni CCS(282) e CCL(283) per leggere lo stato dei flag di condizione da un punto diverso del programma, ad esempio da un altro task.

Nota Quando lo stato dei flag di condizione viene monitorato da una Console di programmazione, sulla Console viene visualizzato lo stato dei flag al termine del ciclo, ovvero lo stato finale dei flag al termine dell'ultimo task del ciclo.

4-2-2 Limitazioni relative alle istruzioni dei task

Istruzioni obbligatorie all'interno dello stesso task

Le istruzioni riportate di seguito devono essere inserite nello stesso task. Qualsiasi tentativo di ripartire le istruzioni tra due task determinerà l'attivazione del flag ER e le istruzioni non verranno eseguite.

Codice mnemonico	Istruzione
JMP/JME	JUMP/JUMP END
CJP/JME	CONDITIONAL JUMP/JUMP END
CJPN/JME	CONDITIONAL JUMP NOT/CONDITIONAL JUMP END
JMP0/JME0	MULTIPLE JUMP/JUMP END
FOR/NEXT	FOR/NEXT
IL/ILC	INTERLOCK/INTERLOCK CLEAR
SBS/SBN/RET	SUBROUTINE CALL/SUBROUTINE ENTRY/SUBROUTINE RETURN
MCRO/SBN/RET	MACRO/SUBROUTINE ENTRY/SUBROUTINE RETURN
BPRG/BEND	BLOCK PROGRAM BEGIN/BLOCK PROGRAM END
STEP S/STEP	STEP DEFINE

Istruzioni non consentite nei task ad interrupt

Non è possibile inserire nei task ad interrupt le istruzioni riportate di seguito. Qualsiasi tentativo di eseguire una di queste istruzioni in un task ad interrupt determinerà l'attivazione del flag ER e l'istruzione non verrà eseguita. È tuttavia possibile eseguire tali istruzioni se il task ad interrupt viene utilizzato come task ciclico supplementare.

Codice mnemonico	Istruzione
TKON(820)	TASK ON
TKOF(821)	TASK OFF
STEP	STEP DEFINE
SNXT	STEP NEXT
STUP	CHANGE SERIAL PORT SETUP
DI	DISABLE INTERRUPT
EI	ENABLE INTERRUPT

L'utilizzo delle seguenti istruzioni in un task ad interrupt potrebbe dare luogo a risultati imprevisti: TIMER: TIM e TIMX((550), HIGH-SPEED TIMER: TIMH(015) e TIMHX(551), ONE-MS TIMER: TMHH(540) e TMHHX(552), ACCUMULATIVE TIMER: TTIM(087) e TTIMX(555), MULTIPLE OUTPUT TIMER: MTIM(543) e MTIMX(554), LONG TIMER: TIML(542) e TIMLX(553), TIMER WAIT: TIMW(813) e TIMWX(816), HIGH-SPEED TIMER WAIT: TMHW(815) e TMHWX(817), PID CONTROL: PID(190), FAILURE POINT DETECTION: FPD(269), CHANGE SERIAL PORT SETUP: STUP(237).

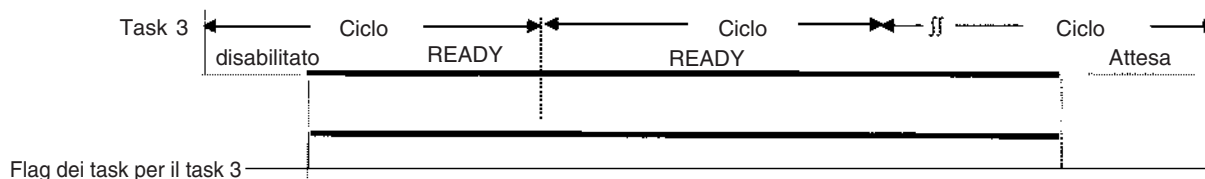
Nel task ad interrupt di spegnimento non è possibile utilizzare le seguenti istruzioni. Le istruzioni non verranno eseguite e **non** verrà attivato il flag di errore: READ DATA FILE: FREAD(700), WRITE DATA FILE: FWRIT(701), NETWORK SEND: SEND(090), NETWORK RECEIVE: RECV(098), DELIVER COMMAND: CMND(490), PROTOCOL MACRO: PMCR(260).

4-2-3 Flag correlati ai task

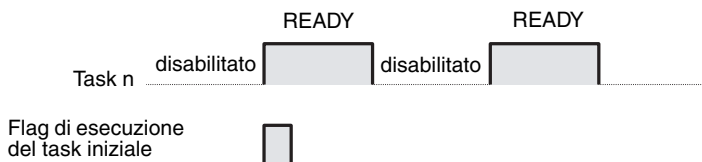
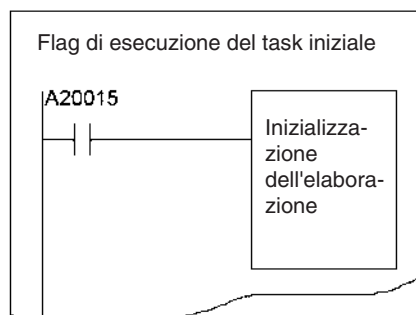
Flag correlati ai task ciclici

I flag riportati di seguito funzionano solo con i task ciclici normali. Non è possibile utilizzare questi flag con i task ciclici supplementari.

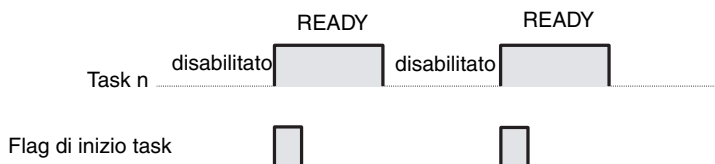
Flag di task (da TK00 a TK31)

Il flag di task viene attivato quando un task ciclico è in stato READY e viene disattivato quando il task è in stato Disabilitato (INI) o in stato di attesa (WAIT). I numeri di task da 00 a 31 corrispondono ai flag di task da TK00 a TK31.

Nota I flag di task vengono utilizzati solo con i task ciclici e non con i task ad interrupt. Per quel che concerne i task ad interrupt, se un task ad interrupt viene eseguito dopo l'avvio del funzionamento, verrà attivato il flag A44115 e il numero del task ad interrupt che richiede il tempo di elaborazione più alto verrà memorizzato in formato esadecimale a due cifre negli indirizzi da A44100 a A44107.

Flag di esecuzione del task iniziale (A20015)

Il flag di esecuzione del task iniziale viene attivato quando lo stato dei task ciclici passa da Disabilitato (INI) a READY, quando i task ottengono l'autorizzazione all'esecuzione e quando vengono eseguiti per la prima volta. Il flag viene disattivato una volta completata la prima esecuzione del task.

Il flag di esecuzione del task iniziale indica se i task ciclici vengono eseguiti per la prima volta. È pertanto possibile utilizzare questo flag per eseguire l'inizializzazione all'interno dei task.

Nota Se un task ciclico in stato di attesa viene riportato allo stato READY utilizzando l'istruzione TKON(820), ciò non viene considerato come esecuzione iniziale e, pertanto, il flag di esecuzione del task iniziale (A20015) non verrà attivato. Il flag di esecuzione del task iniziale (A20015) non verrà altresì attivato se lo stato di un task ciclico passa da Disabilitato a RUN o se un task ciclico viene posto in stato di attesa da un altro task mediante l'istruzione TKOF(821) prima che venga concessa l'autorizzazione all'esecuzione.

Flag di inizio task (A20014) (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)

È possibile utilizzare il flag di inizio task per eseguire l'inizializzazione ogni volta che viene avviato il ciclo del task. Il flag di inizio task viene attivato ogni volta che un task ciclico passa dallo stato Disabilitato (INI) o di attesa (WAIT) allo stato READY, mentre il flag di esecuzione del task iniziale viene attivato solo quando lo stato del task passa da Disabilitato (INI) a READY.

È possibile utilizzare il flag di inizio task per eseguire l'inizializzazione quando un task passa dallo stato di attesa allo stato RUN, ovvero quando un task in attesa viene attivato utilizzando l'istruzione TKON(820).

Flag correlati a tutti i task

Flag di errore di esecuzione task (A29512)

Il flag di errore di esecuzione task viene attivato quando si verifica uno dei seguenti errori:

- Nessun task ciclico (inclusi i task ciclici supplementari) è in stato READY durante un ciclo.
- Il programma assegnato a un task ciclico (inclusi i task ciclici supplementari) non esiste. Questa circostanza non si verifica quando si utilizza CX-Programmer o una Console di programmazione.
- Nessun programma è assegnato a un task ad interrupt attivato.

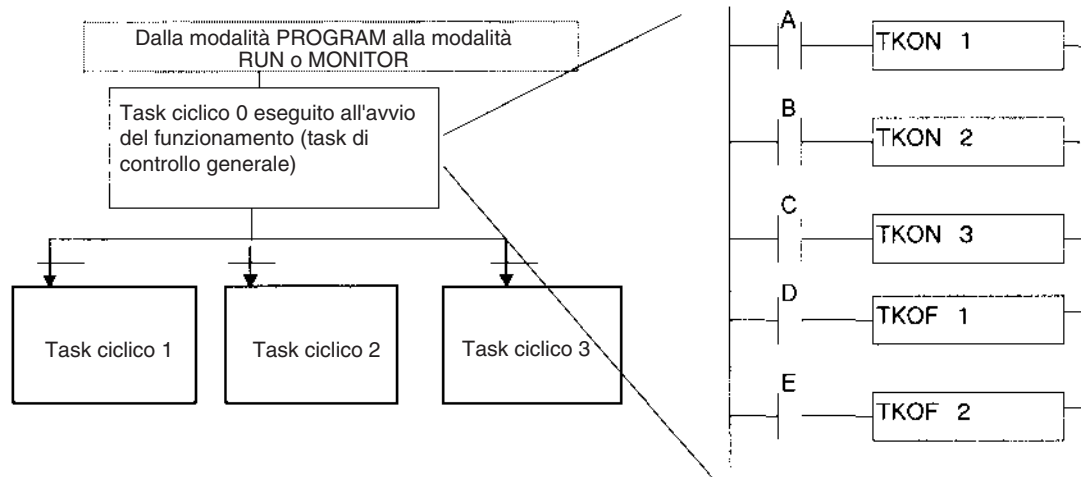
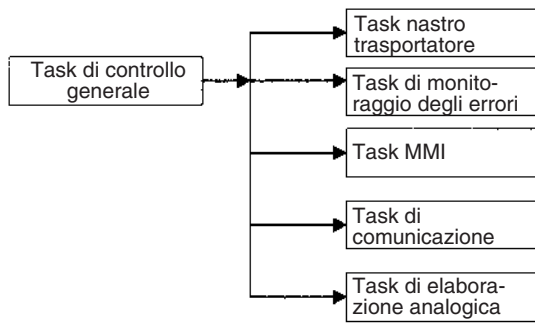
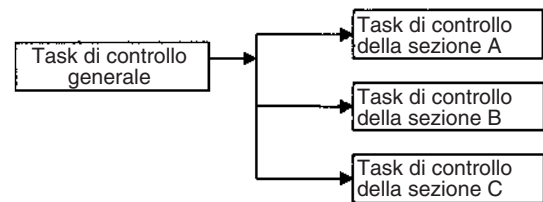
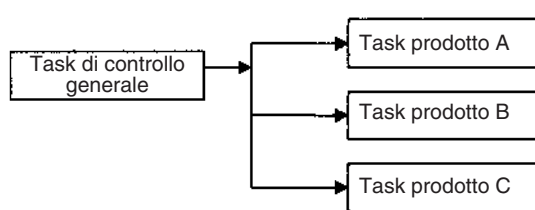
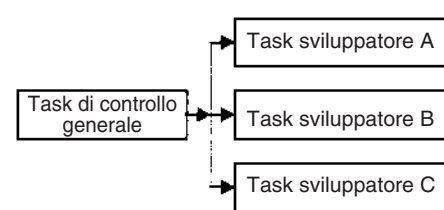
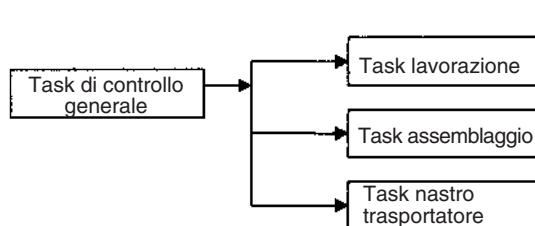
Numero del task all'interruzione del programma (A294)

Quando l'esecuzione di un task viene interrotta a causa di un errore di programma, il tipo e il numero del task corrente verranno memorizzati come segue.

Tipo	A294
Task ciclico	Da 0000 a 001F esadecimale (corrisponde ai numeri di task da 0 a 31)
Task ad interrupt	Da 8000 a 80FF esadecimale (corrisponde ai numeri di task da 0 a 255)

Queste informazioni agevolano l'individuazione del punto in cui si è verificato l'errore fatale e vengono cancellate quando viene rimossa la condizione di errore. L'indirizzo di programma in corrispondenza del quale l'esecuzione del task si è interrotta viene memorizzato in A298 (bit all'estrema destra dell'indirizzo di programma) e in A299 (bit all'estrema sinistra dell'indirizzo di programma).

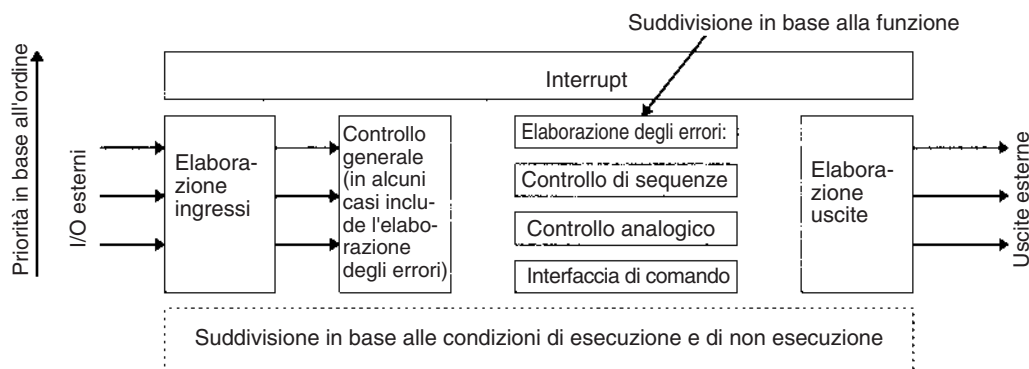
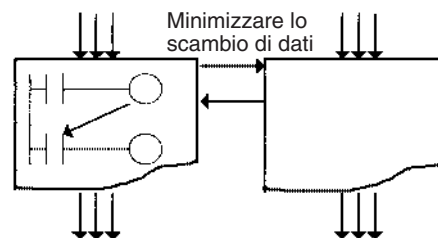
Esempi di task

Per controllare lo stato WAIT/READY di tutti i task ciclici normali e supplementari, viene in genere utilizzato un task di controllo generale impostato per il passaggio allo stato READY all'avvio del funzionamento. Naturalmente, qualsiasi task ciclico è in grado di controllare lo stato WAIT/READY di qualsiasi altro task ciclico in base alle necessità dell'applicazione.

È inoltre possibile combinare le classificazioni sopra illustrate, ad esempio la classificazione per funzione e quella per task.

4-2-4 Progettazione di task

Per la progettazione dei task, si consiglia di attenersi alle indicazioni riportate di seguito.

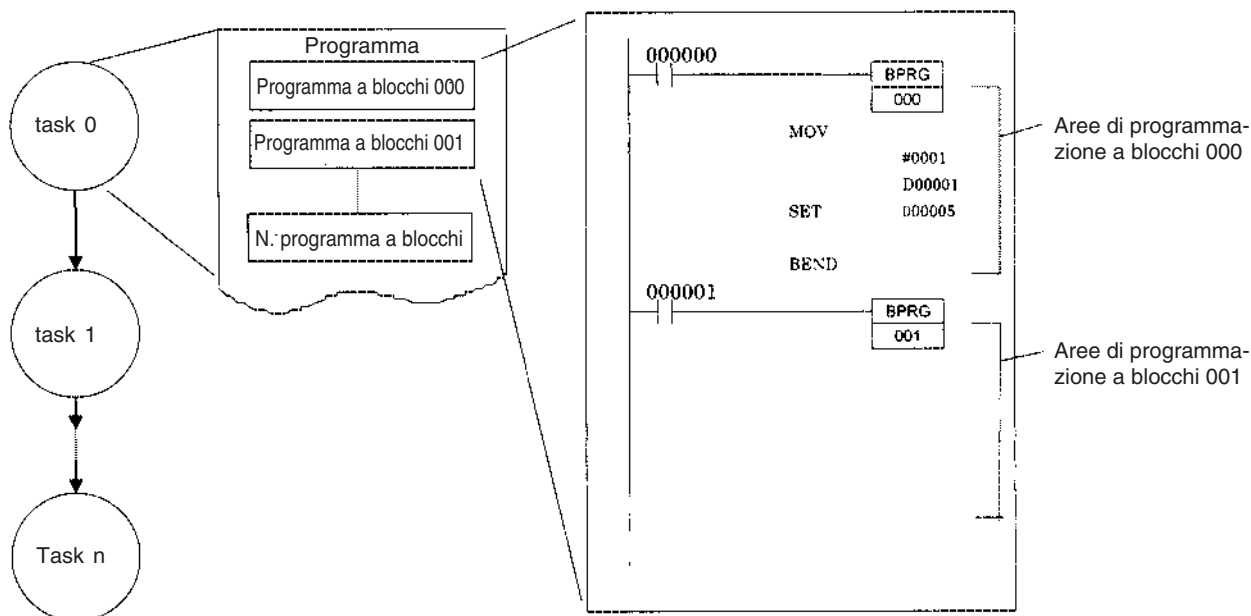
1,2,3...

1. Utilizzare i seguenti standard per determinare la suddivisione dei task:
   a) Riepilogare le condizioni di esecuzione specifiche per i task eseguibili e non eseguibili.
   b) Annotare la presenza o l'assenza di I/O esterni.
   c) Riepilogare le funzioni.
      Ridurre al minimo la quantità di dati scambiati tra i task per il controllo di sequenze, il controllo analogico, l'interfaccia di comando, l'elaborazione di errori e altri task al fine di garantire un elevato livello di autonomia.
   d) Riepilogare l'esecuzione in ordine di priorità.
      Distribuire l'elaborazione tra task ciclici e task ad interrupt.

2. Assicurarsi di suddividere e progettare i programmi in modo da mantenere un elevato grado di autonomia e ridurre al minimo la quantità di dati scambiati tra i task (programmi).

3. Utilizzare un task di controllo generale per controllare lo stato WAIT/READY degli altri task.
4. Assegnare i numeri di task più bassi ai task con priorità più elevata.
   Ad esempio, assegnare un numero più basso al task di controllo anziché ai task di elaborazione.
5. Assegnare numeri di task più bassi ai task ad interrupt con priorità elevata.
6. Un task in stato READY verrà eseguito nei cicli successivi purché il task non passi autonomamente allo stato di attesa o non venga posto in tale stato da un altro task. Se è necessario suddividere l'elaborazione in rami, assicurarsi di utilizzare un'istruzione TKOF(821) (TASK OFF).
7. Utilizzare il flag di esecuzione del task iniziale (A20015) o il flag di inizio task (A20014) nella condizione di esecuzione per le istruzioni di esecuzione al fine di inizializzare i task. Il flag di esecuzione del task iniziale viene attivato alla prima esecuzione di ogni task. Il flag di inizio task viene attivato ogni volta che un task passa allo stato READY.

8. Suddividere la memoria I/O in memoria condivisa dai task e memoria utilizzata solo per task individuali, quindi raggruppare per task la memoria I/O utilizzata solo per task individuali.

Relazione tra task e programmi a blocchi

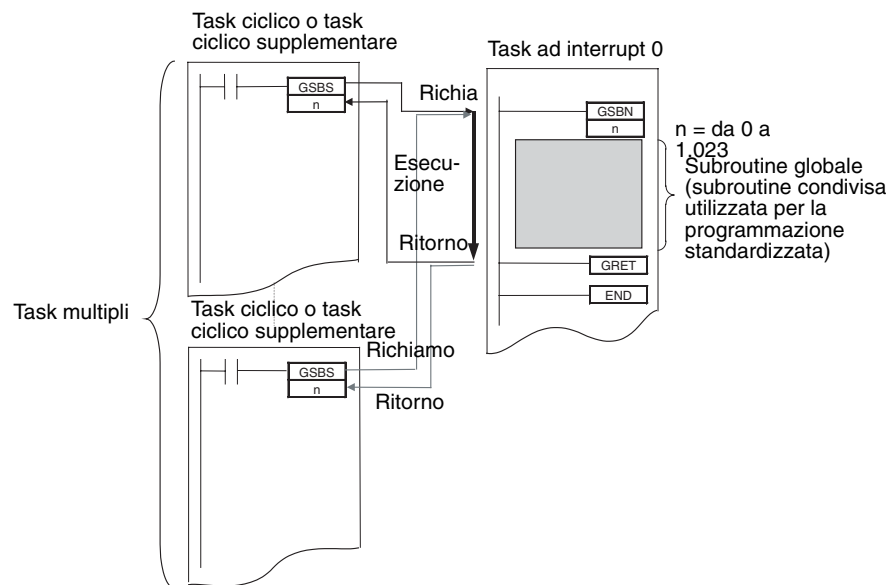
È possibile creare nei task fino a 128 programmi a blocchi. Tale numero rappresenta il numero totale per tutti i task. L'esecuzione di ciascun programma a blocchi è controllata dal diagramma ladder, ma le istruzioni all'interno del programma vengono scritte utilizzando codici mnemonici. In altri termini, un programma a blocchi viene creato combinando un'istruzione ladder e il codice mnemonico.

L'utilizzo di un programma a blocchi semplifica le operazioni di scrittura del flusso logico qualora occorra utilizzare rami condizionali e suddividere i task in step, la cui creazione può risultare difficoltosa tramite l'uso di diagrammi ladder. I programmi a blocchi occupano il livello più basso della gerarchia di programmazione. È infatti possibile suddividere le unità di programma di maggiori dimensioni, rappresentate dai task, in unità di dimensioni minori costituite da programmi a blocchi che utilizzano la stessa condizione di esecuzione (condizione di attivazione).

4-2-5 Subroutine globali

È possibile richiamare le subroutine globali da più task. Le subroutine globali sono supportate solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Per le CPU CS1 o CJ1, non è possibile richiamare una subroutine all'interno di un task da altri task. Tuttavia, nel caso delle CPU CS1-H, CJ1-H, CJ1M e CS1D, è possibile creare subroutine globali nel task ad interrupt numero 0 e tali subroutine possono essere richiamate da task ciclici (task ciclici supplementari inclusi).

Per richiamare una subroutine globale, viene utilizzata l'istruzione GSBS. È necessario che il numero di subroutine sia compreso tra 0 e 1.023. La subroutine globale viene definita alla fine del task ad interrupt numero 0, immediatamente prima dell'istruzione END(001), tra le istruzioni GSBN e GRET.

È possibile utilizzare le subroutine globali per creare una libreria di sezioni di programma standard richiamabili all'occorrenza.

4-3 Task ad interrupt

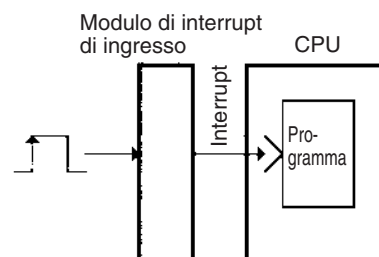
4-3-1 Tipi di task ad interrupt

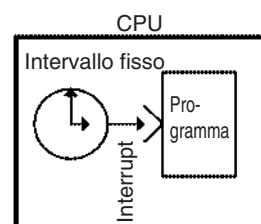
I task ad interrupt possono venire eseguiti in qualsiasi momento del ciclo nelle circostanze descritte di seguito.

Per attivare i task ad interrupt, è possibile utilizzare gli ingressi di interrupt integrati e gli ingressi del contatore ad alta velocità di una CPU CJ1M. Per ulteriori informazioni, fare riferimento al *Manuale dell'operatore degli I/O integrati della serie CJ*.

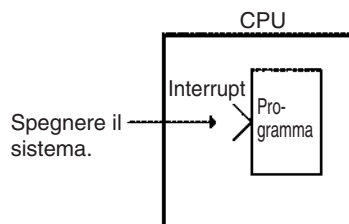
Nota Le CPU CS1D per sistemi a due CPU non supportano gli interrupt. È tuttavia possibile utilizzare i task ad interrupt solo come task ciclici supplementari.

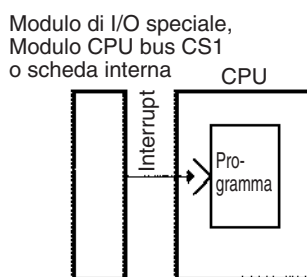
Interrupt di I/O (solo serie CS)

Il task ad interrupt di I/O viene eseguito quando viene attivato un ingresso del Modulo di interrupt di ingresso.

Interrupt programmati

Un task ad interrupt programmato viene eseguito a intervalli regolari.

Interrupt di spegnimento

Il task ad interrupt di spegnimento viene eseguito quando l'alimentazione viene interrotta.

Nota È necessario che il tempo di esecuzione del task ad interrupt di spegnimento sia inferiore a 10 ms meno il ritardo di rilevamento della caduta di tensione.

Interrupt esterni (solo serie CS)

Un task ad interrupt esterno viene eseguito quando un interrupt richiesto da un Modulo di I/O speciale, un Modulo CPU bus o una scheda interna (solo serie CS). Per richiedere l'esecuzione di un task ad interrupt esterno, è tuttavia necessario che il Modulo di I/O speciale o il Modulo CPU bus sia installato nel sistema CPU.

Elenco dei task ad interrupt

Tipo	N. task	Condizione di esecuzione	Procedura di impostazione	Numero di interrupt	Esempi di applicazione
Interrupt di I/O da 00 a 31	Da 100 a 131	Attivazione dell'ingresso del Modulo di interrupt di ingresso installato nel sistema CPU (vedere nota 1)	Utilizzare l'istruzione MSKS (SET INTERRUPT MASK) per assegnare gli ingressi dei Moduli di interrupt di ingresso nel sistema CPU.	32 punti	Aumento della velocità della risposta a ingressi specifici
Interrupt program-mati 0 e 1	2 e 3	Programmati a intervalli regolari	Utilizzare l'istruzione MSKS (SET INTERRUPT MASK) per impostare l'intervallo degli interrupt. Vedere le unità di tempo per gli inter-rupt programmati impostate nella configurazione del PLC.	2 punti	Monitoraggio dello stato operativo a intervalli regolari
Interrupt di spegni-mento	1	All'interruzione dell'alimen-tazione (dopo il tempo di rilevamento della caduta di tensione predefinito + il ritardo di rilevamento della caduta di tensione)	Vedere le impostazioni per il task ad interrupt di spegni-mento e il ritardo di rileva-mento della caduta di tensione nella configurazione del PLC.	1 punto	Esecuzione dell'elaborazione di emergenza in caso di cadute di alimentazione
Interrupt esterni da 0 a 255	da 0 a 255	Su richiesta di un Modulo di I/O speciale o un Modulo CPU bus instal-lato nel sistema CPU o di una scheda interna (solo serie CS) (vedere nota 2)	Nessuna (sempre abilitati)	256 punti	Esecuzione dell'elabora-zione richiesta da un Modulo di I/O speciale, un Modulo CPU bus o dalla scheda interna

Nota

1. È necessario che il Modulo di interrupt di ingresso sia installato nel sistema CPU. Nel caso delle CPU CJ1-H è necessario collegare tale Modulo in prossimità della CPU, in uno dei cinque slot (da 0 a 4) disponibili. Nel caso delle CPU CJ1M è necessario collegare tale Modulo in prossimità della CPU, in uno dei tre slot (da 0 a 2) disponibili. Per richiedere l'esecuzione

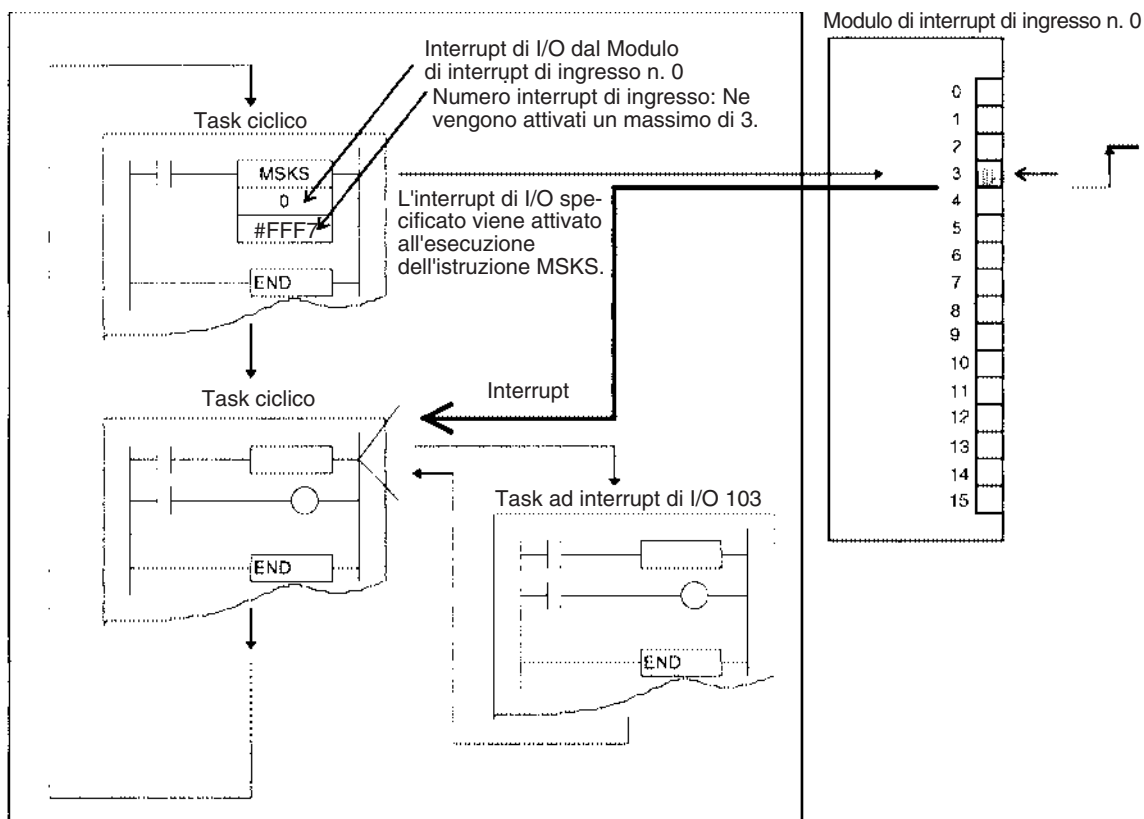
dei task ad interrupt di I/O, non è possibile utilizzare Moduli di interrupt di I/O installati in posizioni diverse da quelle indicate.

2. Il Modulo di I/O speciale o il Modulo CPU bus deve essere installato nel sistema CPU. Nel caso delle CPU CJ1-H è necessario collegare tale Modulo in prossimità della CPU, in uno dei cinque slot (da 0 a 4) disponibili. Nel caso delle CPU CJ1M è necessario collegare tale Modulo in prossimità della CPU, in uno dei tre slot (da 0 a 2) disponibili. Per generare interrupt esterni, non è possibile utilizzare Moduli installati in posizioni diverse da quelle indicate.
3. Le CPU CJ1 per sistemi a due CPU non supportano task ad interrupt di I/O e task ad interrupt esterni.
4. Le CPU CS1D per sistemi a due CPU non supportano gli interrupt. Con le CPU CS1D è possibile utilizzare i task ad interrupt solo come task ciclici supplementari. In pratica, non è possibile utilizzare alcun altro tipo di task ad interrupt.

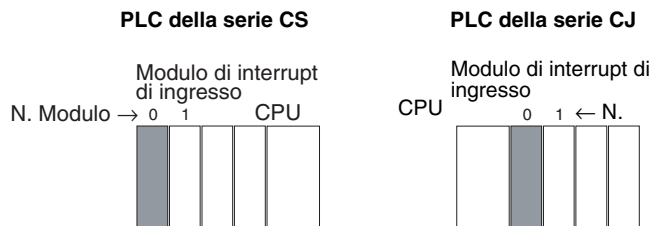
Task ad interrupt di I/O: task da 100 a 131

Per impostazione predefinita, i task ad interrupt di I/O vengono disabilitati all'avvio dell'esecuzione dei task ciclici. Per abilitare gli interrupt di I/O, eseguire l'istruzione MSKS (SET INTERRUPT MASK) in un task ciclico per il numero di interrupt del Modulo di interrupt di ingresso.

Ad esempio, L'esempio fornito di seguito illustra l'esecuzione del task ad interrupt di I/O 103 quando l'ingresso numero 3 del Modulo di interrupt di ingresso 0 (il Modulo all'estrema sinistra tra 0 e 1) viene attivato.

Nota Non abilitare task ad interrupt di I/O non necessari. Se l'interrupt di ingresso viene generato da disturbi e non esiste un task ad interrupt corrispondente, si verificherà un errore fatale (errore di task) che causerà l'interruzione del programma.

Numeri di Modulo di interrupt di ingresso, numeri di ingresso e numeri di task ad interrupt di I/O

N. Modulo di interrupt di ingresso (vedere nota)	N. ingresso	Task ad interrupt di I/O
0	da 0 a 15	**Da 100 a 115**
1	da 0 a 15	**Da 116 a 131**

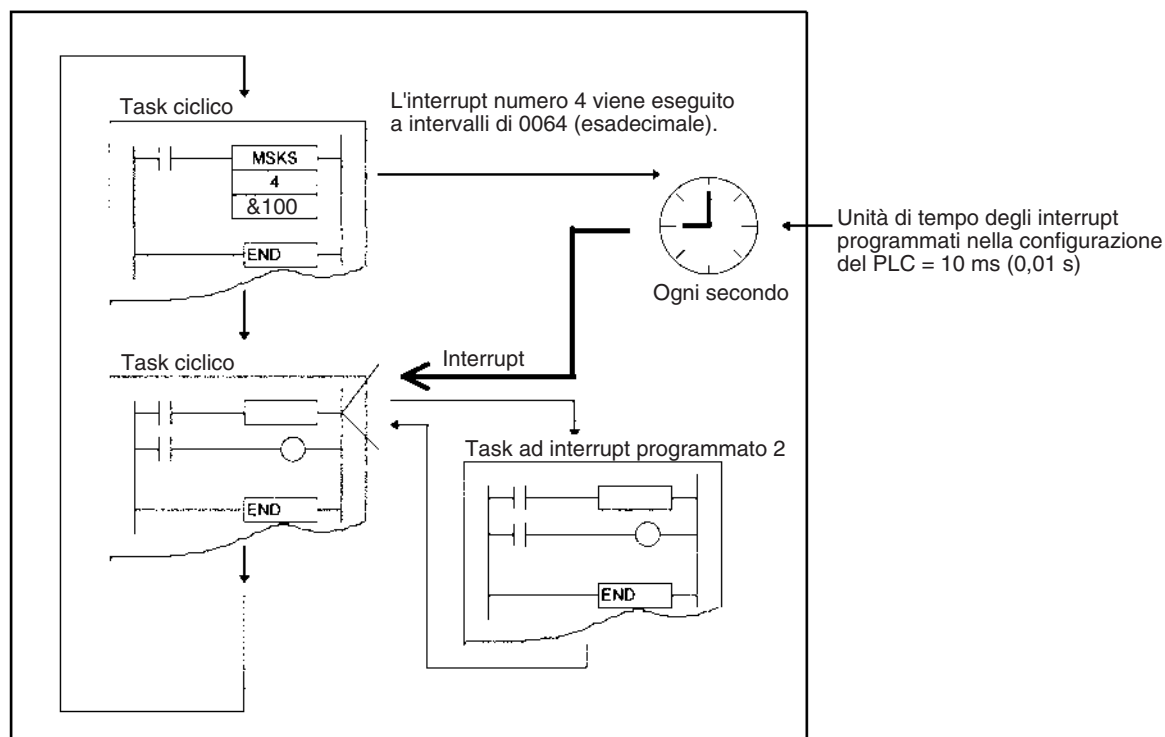
Nota Per i PLC della serie CS, i Moduli di interrupt di ingresso sono numerati da 0 a 1, a partire dal lato sinistro del sistema CPU. Per i PLC della serie CJ, i Moduli di interrupt di ingresso sono numerati da 0 a 1, a partire dalla CPU.

Operando S (secondo operando) di MSKS. I bit di FFF7 esadecimale corrispondono agli ingressi del Modulo di interrupt di ingresso. I numeri di interrupt di ingresso da 0 a 15 corrispondono ai bit da 0 a 15.

Task ad interrupt programmati: task 2 e 3

Nella configurazione del PLC predefinita, i task ad interrupt programmati risultano disabilitati all'avvio dell'esecuzione dei task ciclici. Per abilitare i task ad interrupt programmati, eseguire le operazioni riportate di seguito.

1,2,3...
1. Eseguire l'istruzione MSKS (SET INTERRUPT MASK) da un task ciclico e impostare il tempo (ciclo) per l'interrupt programmato specificato.
2. Impostare l'unità di tempo per l'interrupt programmato nella configurazione del PLC.

Nota L'impostazione del tempo per gli interrupt incide sul task ciclico con le seguenti considerazioni: quanto più breve è l'intervallo di interrupt, maggiore sarà la frequenza di esecuzione del task e più lungo risulterà il tempo di ciclo.

Esempio L'esempio fornito di seguito illustra l'esecuzione del task ad interrupt programmato 2 ogni secondo.

Numeri di interrupt e numero di task ad interrupt programmato

N. interrupt	Task ad interrupt programmato
4	2
5	3

Impostazioni di configurazione del PLC

Indirizzo	Nome	Descrizione	Impostazioni	Impostazione predefinita
Bit da 0 a 3 di 195	Unità di tempo degli interrupt programmati	Imposta l'unità di tempo per gli interrupt programmati in modo che i task ad interrupt vengano eseguiti a intervalli regolari.	00 esadecimale: 10 ms 01 esadecimale: 1,0 ms 02 esadecimale: 0,1 ms (solo CPU CJ1M)	00 esadecimale

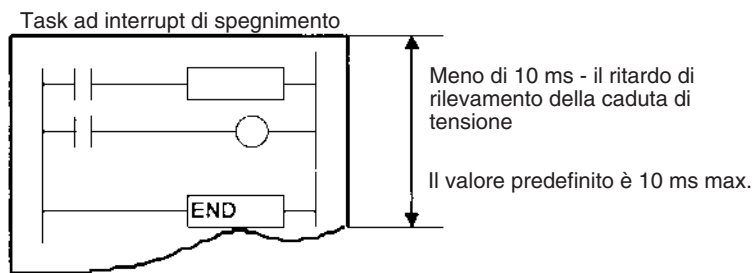
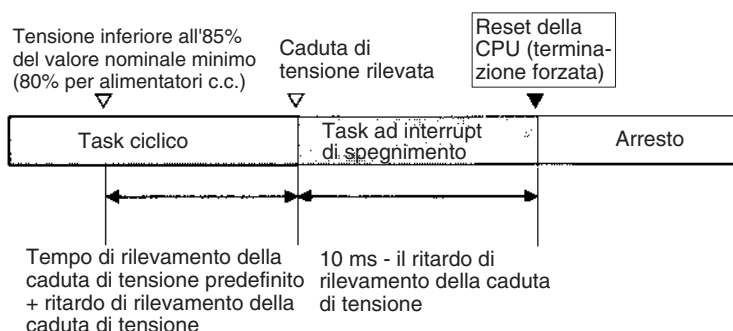
Task ad interrupt di spegnimento: task 1

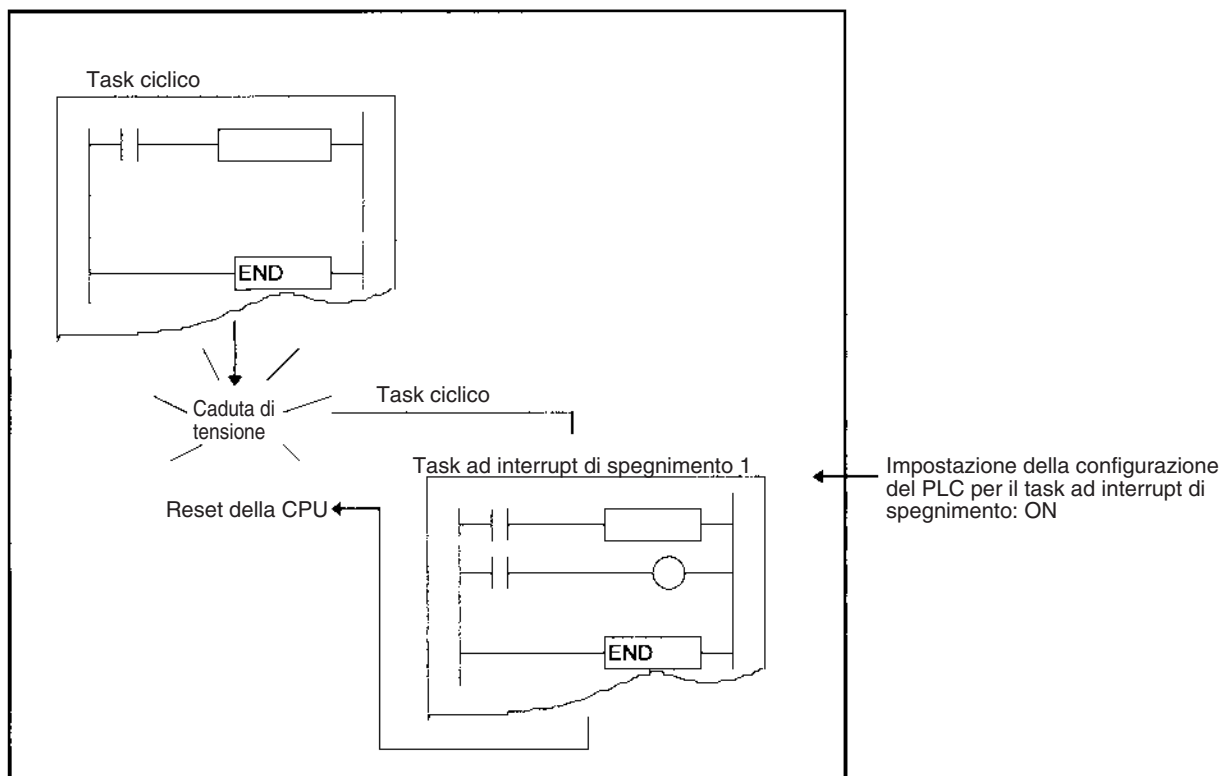
Nella configurazione del PLC predefinita, il task ad interrupt di spegnimento risulta disabilitato all'avvio dell'esecuzione dei task ciclici.

È possibile abilitare il task ad interrupt di spegnimento nella configurazione del PLC.

In base alla configurazione del PLC predefinita, il task ad interrupt di spegnimento viene interrotto dopo 10 ms. È pertanto necessario che il tempo di esecuzione del task ad interrupt di spegnimento sia inferiore a 10 ms.

Se nella configurazione del PLC viene impostato un ritardo di rilevamento della caduta di tensione, il task ad interrupt di spegnimento verrà arrestato dopo 10 ms meno il valore impostato per il ritardo. In questo caso, il tempo di esecuzione del task ad interrupt di spegnimento deve essere inferiore a 10 ms meno il valore impostato per il ritardo di rilevamento della caduta di tensione nella configurazione del PLC.

Ad esempio, Se il ritardo di rilevamento della caduta di tensione viene impostato su 4 ms nella configurazione del PLC, il tempo di esecuzione deve essere inferiore a 10 ms meno 4 ms, ovvero 6 ms.

Nota Una condizione di spegnimento viene riconosciuta come tale quando l'alimentazione scende al di sotto dell'85% della tensione nominale minima (80% per gli alimentatori c.c.), e il tempo disponibile per l'elaborazione prima che il task ad interrupt di spegnimento venga eseguito è rappresentato dal tempo di rilevamento della caduta di tensione predefinito (da 10 a 25 ms per gli alimentatori c.a. e da 2 a 5 ms per gli alimentatori c.c.) più il ritardo di rilevamento della caduta di tensione impostato nella configurazione del PLC (da 0 a 10 ms). I task ciclici verranno eseguiti per la durata di questo periodo.

Nota Accertarsi che il tempo di esecuzione del task ad interrupt di spegnimento sia inferiore a 10 ms meno il valore impostato per il ritardo di rilevamento della caduta di tensione nella configurazione del PLC. Allo scadere di questo periodo di tempo, le eventuali istruzioni rimanenti non verranno eseguite. Se l'alimentazione viene interrotta durante la modifica in linea, il task ad interrupt di spegnimento non verrà eseguito. Oltre alle istruzioni non consentite in alcun task ad interrupt (per informazioni dettagliate, fare riferimento al manuale *Instructions Reference Manual*), nel task ad interrupt di spegnimento non è possibile utilizzare le seguenti ulteriori istruzioni: READ DATA FILE: FREAD(700), WRITE DATA FILE: FWRIT(701), NETWORK SEND: SEND(090), NETWORK RECEIVE: RECV(098), DELIVER COMMAND: CMND(490), TRANSMIT: TXD(236), RECEIVE: RXD(235), PROTOCOL MACRO: PMCR(260).

Esecuzione del task ad interrupt di spegnimento

Impostazioni della configurazione del PLC per il task ad interrupt di spegnimento (numero di task 1)

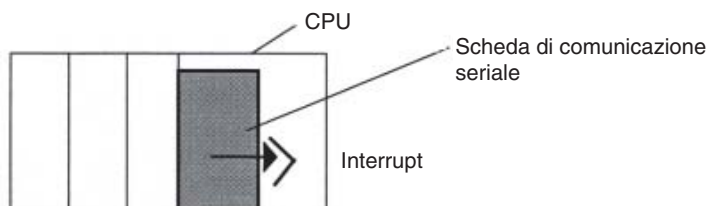
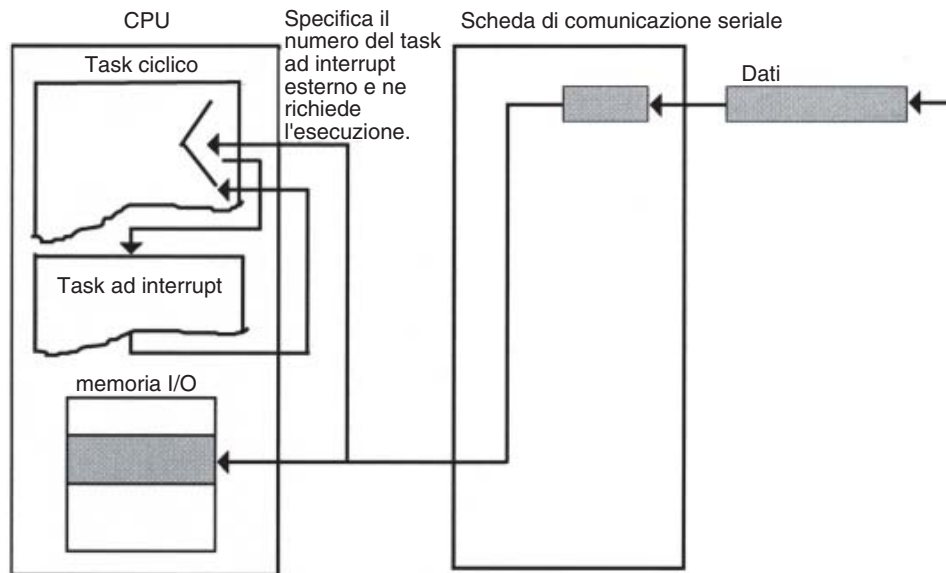
Indirizzo	Nome	Descrizione	Impostazioni	Impostazione predefinita
Bit 15 di +225	Abilitazione/disabilitazione del task ad interrupt di spegnimento	Se il bit 15 di +225 è impostato su ON, il task ad interrupt di spegnimento verrà avviato in caso di interruzione dell'alimentazione.	0: OFF; 1: ON	0
Bit da 0 a 7 di +225	Ritardo di rilevamento della caduta di tensione	La condizione di spegnimento viene riconosciuta alla scadenza del periodo dato dal ritardo di rilevamento della caduta di tensione più il tempo di rilevamento della caduta di tensione predefinito (da 10 a 25 ms per gli alimentatori c.a. e da 2 a 5 ms per gli alimentatori c.c.).	Da 00 a 0A esadecimale: da 1 a 10 ms (unità: 0 ms)	00 esadecimale

Task ad interrupt esterni: task da 0 a 255

La ricezione di un task ad interrupt esterno può avvenire in qualsiasi momento.

L'elaborazione degli interrupt viene eseguita nella CPU dei PLC contenenti una scheda interna (solo serie CS), Moduli di I/O speciale o Moduli CPU bus. Non è necessario configurare alcuna impostazione a livello di CPU a meno che il programma non contenga un task ad interrupt esterno per un numero di task specifico.

Le CPU CJ1-H non supportano gli interrupt esterni.

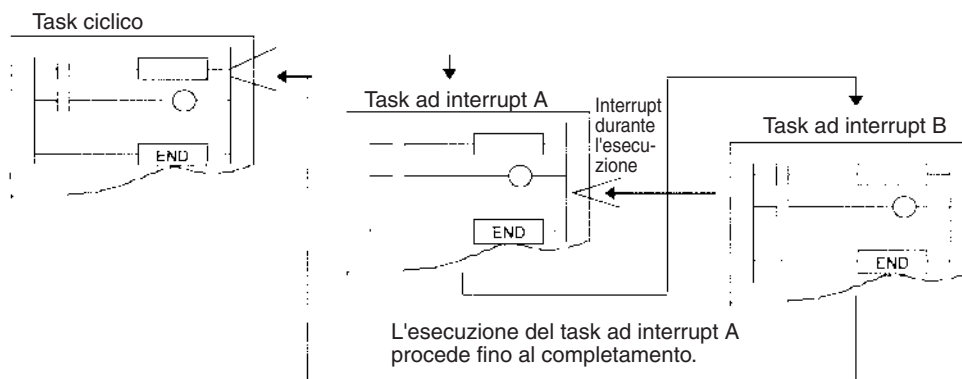
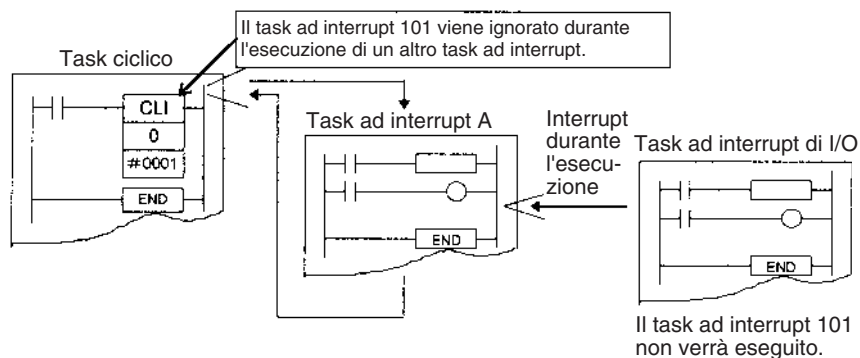
Ad esempio, L'esempio fornito di seguito illustra un interrupt esterno generato da una Scheda di comunicazione seriale CS1W-SCB□1.

Quando il metodo di notifica della risposta della Scheda di comunicazione seriale è impostato per la notifica di interrupt in base a numero fisso o la notifica di interrupt in base a numero ricezione, la Scheda invierà alla CPU la richiesta di esecuzione di un task ad interrupt esterno dopo avere ricevuto i dati dalla relativa porta seriale e scriverà tali dati nella memoria I/O della CPU.

Nota

1. Quando il metodo di notifica della risposta viene impostato per la notifica di interrupt in base a numero fisso, la Scheda richiederà l'esecuzione del task ad interrupt avente il numero di task preimpostato.
2. Quando il metodo di notifica della risposta viene impostato per la notifica di interrupt in base a numero ricezione, il numero del task ad interrupt esterno verrà calcolato in base alla formula specificata e la Scheda richiederà l'esecuzione del task ad interrupt avente il numero di task calcolato.
3. Se il numero di un task ad interrupt esterno (da 0 a 255) coincide con quello di un task ad interrupt di spegnimento (task 1), un task ad interrupt programmato (task 2 o 3) o un task ad interrupt di I/O (da 100 a 131), verrà eseguito il task associato a una delle due condizioni di interrupt (interrupt esterno o l'altra condizione di interrupt). Di norma, è consigliabile non assegnare numeri di task duplicati.

4-3-2 Priorità dei task ad interrupt

Per consentire l'esecuzione del task ad interrupt di spegnimento, viene interrotta l'esecuzione di un altro task ad interrupt. La CPU verrà ripristinata ma il task ad interrupt interrotto non verrà eseguito in seguito all'esecuzione del task ad interrupt di spegnimento.

Interrupt durante l'esecuzione dei task ad interrupt

Se si verifica una condizione di interrupt quando è in esecuzione un altro task ad interrupt, il task relativo all'interrupt ricevuto non verrà eseguito fino al completamento di quello originale.

Nota Se non si desidera che un numero di task ad interrupt di I/O specifico venga salvato ed eseguito per una CPU della serie CS qualora si verifichi durante l'esecuzione un altro task ad interrupt, eseguire l'istruzione CLI (CLEAR INTERRUPT) dall'altro task ad interrupt per cancellare il numero di interrupt salvato internamente. Non è possibile annullare gli interrupt programmati e gli interrupt esterni.

Generazione simultanea di più interrupt

Quando vengono ricevuti più interrupt contemporaneamente, i task ad interrupt diversi dai task ad interrupt di spegnimento vengono eseguiti nel seguente ordine di priorità:

Task ad interrupt di I/O (solo serie CS) > task ad interrupt esterni (solo serie CS) > task ad interrupt programmati

Se vi sono più interrupt per tipo, i task verranno eseguiti a partire dal task con numero più basso.

Nota Per ogni task ad interrupt viene registrato in memoria un solo interrupt, mentre per un task ad interrupt già in esecuzione non viene registrato alcun interrupt. A causa della bassa priorità assegnata agli interrupt programmati e poiché viene registrato un solo interrupt per volta, è possibile che un interrupt programmato venga omesso.

4-3-3 Canali e flag dei task ad interrupt

Tempo di elaborazione massimo del task ad interrupt (A440)

Il tempo di elaborazione massimo per un task ad interrupt viene memorizzato come dato binario in unità di 0,1 ms e viene azzerato all'avvio del funzionamento.

Task ad interrupt con tempo di elaborazione più alto (A441)

Il numero del task ad interrupt che richiede il tempo di elaborazione più alto viene memorizzato come dato binario. In questo caso, i valori esadecimali da 8000 a 80FF corrispondono ai numeri di task da 00 a FF esadecimale.

Il flag A44115 viene attivato alla ricezione del primo interrupt dopo l'avvio del funzionamento. Il tempo di elaborazione massimo per i task ad interrupt successivi verrà memorizzato in formato esadecimale nelle due cifre all'estrema destra e verrà azzerato all'avvio del funzionamento.

Flag di errore task ad interrupt (errore non fatale) (A40213)

Se la funzione di rilevamento degli errori dei task ad interrupt è attivata nella configurazione del PLC, il flag di errore task ad interrupt verrà attivato in caso di errore.

Flag di errore task ad interrupt (A42615)/numero del task che genera l'errore del task ad interrupt (da A42600 a 42611)

Se viene attivato il flag A40213, i dati riportati di seguito verranno memorizzati in A42615 e nell'intervallo da A42600 ad A42611.

A40213	Descrizione errore del task ad interrupt	A42615	Da A42600 a 42611
Errore del task ad interrupt (se la funzione di rilevamento degli errori dei task ad interrupt è attivata nella configurazione del PLC)	Si verifica se un task ad interrupt viene eseguito per più di 10 ms durante l'aggiornamento del Modulo di I/O speciale C200H o del Modulo di I/O remoto SYSMAC BUS (solo serie CS).	OFF	Il numero del task ad interrupt verrà memorizzato in 12 bit di dati binari (task ad interrupt da 0 a 255: da 000 a FFF esadecimale).
	Si verifica durante un tentativo di aggiornamento degli I/O per un numero elevato di canali da un task ad interrupt utilizzando l'istruzione IORF mentre un Modulo di I/O speciale è già in fase di aggiornamento ciclico.	ON	Il numero del Modulo di I/O speciale in fase di aggiornamento verrà memorizzato in 12 bit di dati binari (numero di modulo da 0 a 95: da 000 a 05F esadecimale).

Numero del task all'interruzione del programma (A294)

Quando l'esecuzione del programma viene interrotta a causa di un errore di programma, il tipo e il numero del task corrente vengono memorizzati nelle seguenti posizioni.

Tipo	A294
Task ad interrupt	Da 8000 a 80FF esadecimale (corrisponde ai numeri di task da 0 a 255)
Task ciclico	Da 0000 a 001F esadecimale (corrisponde ai numeri di task da 0 a 31)

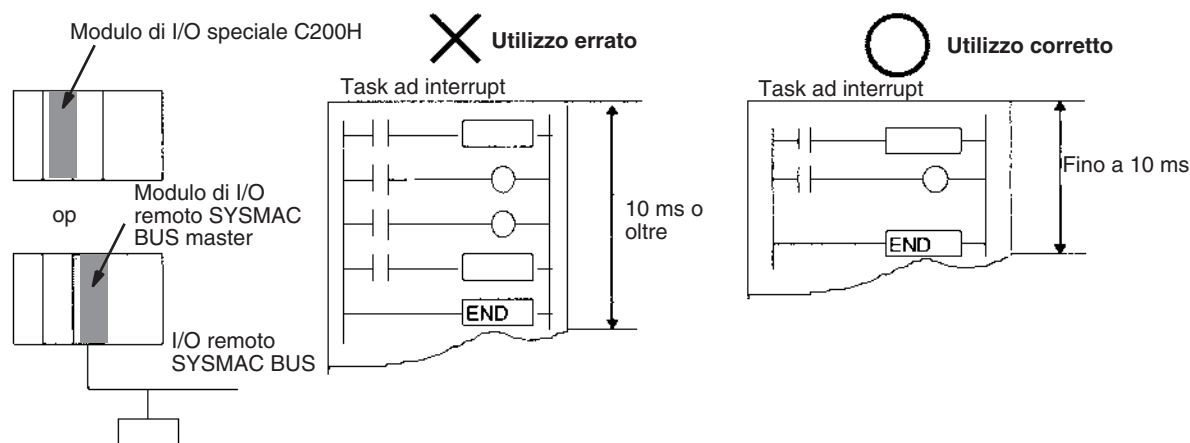
4-3-4 Precauzioni relative all'applicazione

Tempi di esecuzione lunghi con Moduli di I/O speciale C200H o SYSMAC BUS (solo serie CS)

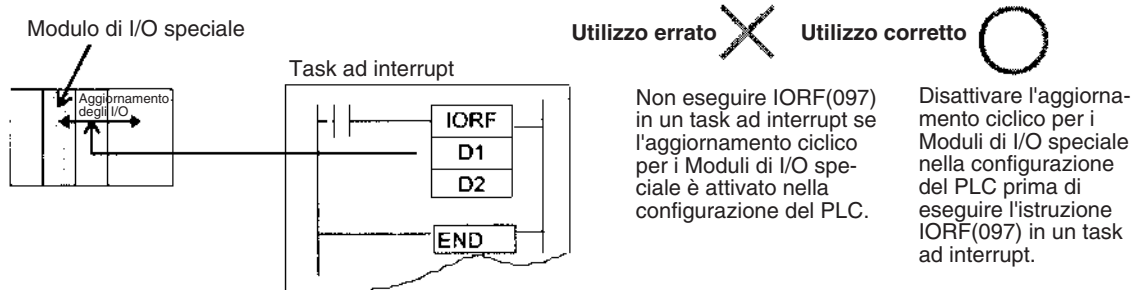
Quando si utilizzano Moduli di I/O speciale o Moduli di I/O remoto SYSMAC BUS, assicurarsi che il tempo di esecuzione di tutti i task ad interrupt (task ad interrupt di I/O, programmati, di spegnimento ed esterni) non sia superiore a 10 ms.

Se un task ad interrupt viene eseguito per oltre 10 ms durante l'aggiornamento di un Modulo di I/O speciale C200H o un Modulo di I/O remoto SYSMAC BUS, si verificherà un errore del task ad interrupt, verrà attivato il flag di errore del Modulo di I/O speciale (A40206) e l'aggiornamento degli I/O del Modulo verrà interrotto. Il funzionamento della CPU non verrà tuttavia interrotto.

Se la funzione di rilevamento degli errori dei task ad interrupt è attivata nella configurazione del PLC, il flag di errore task ad interrupt (A40213) verrà attivato quando si verifica un errore e il numero del task ad interrupt che ha

generato l'errore verrà memorizzato in A426 (errore del task ad interrupt e numero di task). Il funzionamento della CPU non verrà tuttavia interrotto.

Esecuzione di IORF per un Modulo di I/O speciale

Se è necessario eseguire un'istruzione IORF(097) da un task ad interrupt per un Modulo di I/O speciale, assicurarsi di disattivare l'aggiornamento ciclico per il Modulo (specificando il numero di modulo) nella configurazione del PLC.

Se si tenta di aggiornare un Modulo di I/O speciale da un task ad interrupt utilizzando un'istruzione IORF(097) mentre tale Modulo è già in fase di aggiornamento, avviato mediante un aggiornamento degli I/O ciclico, un'istruzione di aggiornamento degli I/O (IORF(097) o istruzioni di aggiornamento immediato (!)), si verificherà un errore del task ad interrupt. Se la funzione di rilevamento degli errori dei task ad interrupt è attivata nella configurazione del PLC quando si verifica l'errore, il flag di errore task ad interrupt (A40213) verrà attivato e il numero del Modulo di I/O speciale per il quale è stato specificato un doppio aggiornamento degli I/O verrà memorizzato in A426 (errore del task ad interrupt e numero di task). Il funzionamento della CPU non verrà interrotto.

Nota Per determinare quale degli errori di task ad interrupt sopra descritti si è verificato, è possibile utilizzare i bit all'estrema sinistra di A426 (errore del task ad interrupt e numero di task). Bit 15: 0 indica un errore associato a un tempo di esecuzione pari o superiore a 10 ms; 1 indica un errore associato alla doppia specifica dell'aggiornamento degli I/O.

Impostazioni di configurazione del PLC

Indirizzo	Nome	Descrizione	Impostazioni	Impostazione predefinita
Bit 14 di +128	Rilevamento degli errori dei task ad interrupt	Specifica se rilevare o meno gli errori dei taska ad interrupt. Quando la funzione di rilevamento è attivata, il flag di errore task ad interrupt (A40213) è operativo.	0: rilevamento attivato 1: rilevamento disattivato	0

Flag e canali dell'area ausiliaria correlati

Nome	Indirizzo	Descrizione
Flag di errore task ad interrupt	A40213	Viene attivato se un task ad interrupt viene eseguito per più di 10 ms durante l'aggiornamento del Modulo di I/O speciale C200H o del Modulo di I/O remoto SYSMAC BUS, tuttavia il funzionamento della CPU non viene interrotto. L'indicatore ERR/ALM sul pannello frontale si accederà (solo serie CS). Viene attivato se si tenta di aggiornare un Modulo di I/O speciale da un task ad interrupt utilizzando un'istruzione IORF mentre tale Modulo è già in fase di aggiornamento ciclico.
Errore del task ad interrupt - Numero task	A426	Contiene il numero del task ad interrupt o del Modulo di I/O speciale in fase di aggiornamento. Il bit 15 viene impostati su OFF quando l'esecuzione di un task ad interrupt richiede 10 ms o tempi più lunghi e viene impostato su ON dopo l'esecuzione del doppio aggiornamento del Modulo di I/O speciale.

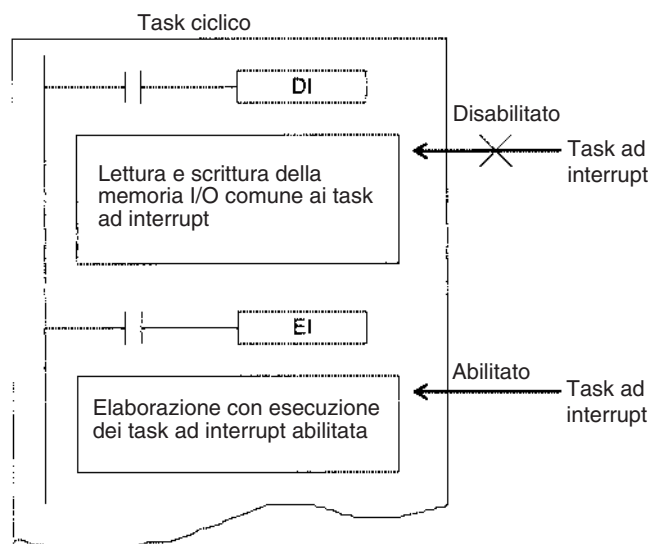
Disabilitazione degli interrupt

L'elaborazione viene interrotta e il task ad interrupt viene eseguito nei seguenti casi:

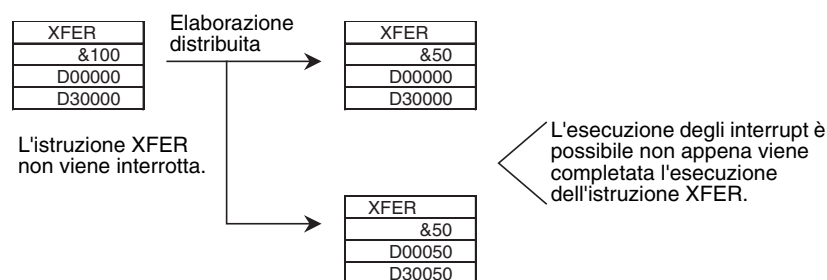
- Durante l'esecuzione di un'istruzione
- Durante l'aggiornamento del Modulo di I/O di base, del Modulo CPU bus, della scheda interna (solo serie CS) o del Modulo di I/O remoto SYSMAC BUS (solo serie CS)
- Durante la gestione delle periferiche tramite Host Link

Corrispondenza dei dati tra task ciclici e ad interrupt

Se un task ciclico (task ciclici supplementari inclusi) e un task ad interrupt eseguono operazioni di lettura e scrittura sugli stessi indirizzi di memoria I/O, i dati potrebbero non corrispondere. Per disabilitare gli interrupt durante l'accesso alla memoria da parte delle istruzioni dei task ciclici, attenersi alla procedura riportata di seguito.

- Inserire un'istruzione DI (DISABLE INTERRUPT) per disabilitare l'esecuzione dei task ad interrupt immediatamente prima dell'operazione di lettura o scrittura di un'istruzione di un task ciclico.
- Inserire un'istruzione EI (ENABLE INTERRUPT) immediatamente dopo il codice di elaborazione per abilitare nuovamente l'esecuzione dei task ad interrupt.

È possibile che si verifichino problemi di corrispondenza dei dati nonostante si utilizzino le istruzioni DI(693) ed EI(694) per disattivare i task ad interrupt durante

l'esecuzione di un'istruzione che richiede la ricezione e l'elaborazione della risposta, quale un'istruzione di rete o un'istruzione di comunicazione seriale.

Nota Nelle CPU CS1-H, CJ1-H, CJ1M e CS1D, l'esecuzione delle istruzioni BIT COUNTER (BCNT), BLOCK SET (BSET) e BLOCK TRANSFER (XFER) non verrà interrotta per consentire l'esecuzione del task ad interrupt, ovvero l'esecuzione dell'istruzione verrà completata prima che il task ad interrupt venga eseguito, ritardando la risposta dell'interrupt. Per evitare che ciò si verifichi, distribuire l'elaborazione dei dati tra più istruzioni, come illustrato di seguito per l'istruzione XFER.

4-4 Operazioni dei dispositivi di programmazione relative ai task

4-4-1 Utilizzo di più task ciclici

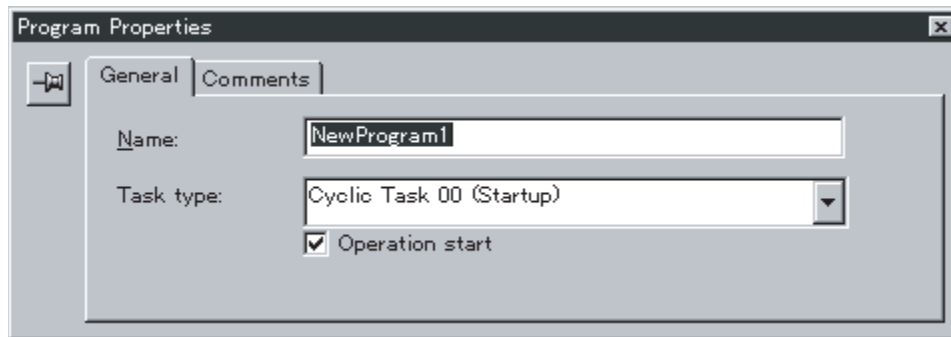
Per creare più task ciclici (task ciclici supplementari inclusi), utilizzare CX-Programmer. Non è possibile utilizzare una Console di programmazione per creare nuovi task ciclici. Per assegnare il tipo di task e il numero di task per i programmi creati, utilizzare CX-Programmer.

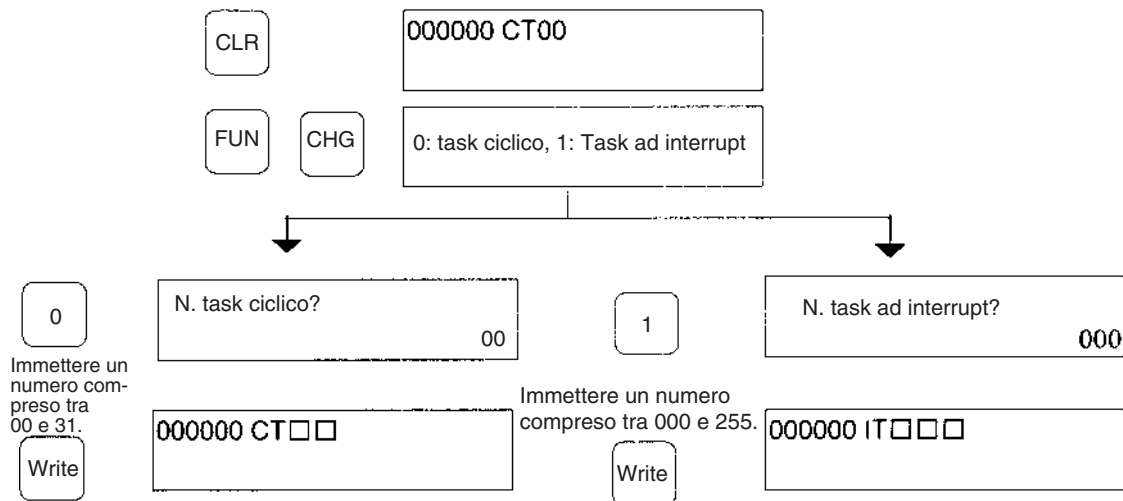
- È possibile utilizzare una Console di programmazione per monitorare o modificare più task ciclici creati e trasferiti a una CPU mediante CX-Programmer.
- La Console di programmazione può essere utilizzata per creare un task ciclico e uno o più task ad interrupt specifici. A tale scopo, è sufficiente utilizzare la funzione di azzeramento totale e specificare i task ad interrupt. La Console di programmazione consente di creare solo i task ad interrupt 1 (interrupt di spegnimento), 2 e 3 (interrupt programmati) e i task da 100 a 131 (interrupt di I/O). Se si utilizza una CPU CJ1M, è possibile creare anche i task ad interrupt da 140 a 143 per gli ingressi integrati. Il task ciclico 0 viene avviato all'avvio del PLC.

4-4-2 Operazioni dei dispositivi di programmazione

CX-Programmer Specificare il tipo e il numero di task come attributi per ciascun programma.

1,2,3...
1. Per visualizzare il programma a cui verrà assegnato un task, selezionare ***View, Properties*** (Visualizza, Proprietà) o fare clic con il pulsante destro del mouse e scegliere **Properties** (Proprietà) dal menu popup.
2. Fare clic sulla scheda **General** (Generale), quindi selezionare il tipo e il numero di task da **Task Type** (Tipo task) e **Task No.** (N. task). Per attivare

il task ciclico, fare clic sulla casella di controllo **Operation start** (Inizio operazione).

Console di programmazione

Sulla Console di programmazione un task viene gestito come fosse un intero programma. Accedere e modificare un programma con una Console di programmazione specificando l'intervallo da CT00 a CT31 per un task ciclico o da IT001 a IT255 per un task ad interrupt.

Nota

1. Non è possibile utilizzare una Console di programmazione per creare nuovi task ciclici.
2. Le CPU della serie CJ1 attualmente non supportano i task ad interrupt di I/O o i task ad interrupt esterni. È possibile specificare solo l'intervallo da IT001 a IT003.

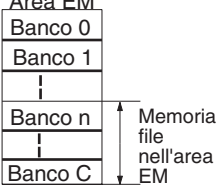
CAPITOLO 5
Funzioni della memoria file

In questo capitolo vengono descritte le funzioni utilizzate per la gestione della memoria file.

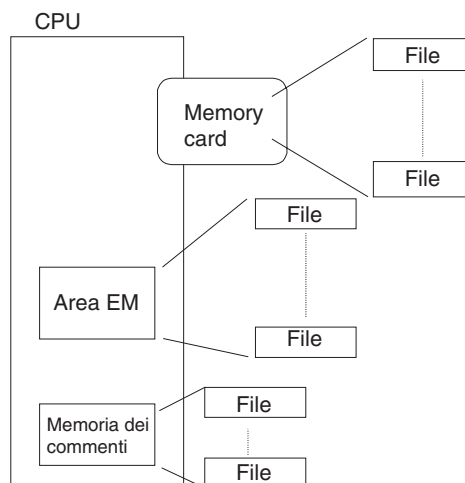
5-1 Memoria file

La serie CS/CJ supporta la memoria file. Per memorizzare i file, è possibile utilizzare i supporti riportati di seguito.

1,2,3...
1. schedine di memoria
2. Un intervallo specifico dell'area EM denominato memoria file nell'area EM

   Nota Poiché le CPU CJ1M non dispongono di un'area EM, non è possibile utilizzare la memoria file nell'area EM.

È possibile utilizzare entrambi i tipi di memoria per archiviare l'intero programma utente, la memoria I/O e le aree dei parametri sotto forma di file.

3. Memoria dei commenti (nella memoria flash interna della CPU)

Per informazioni dettagliate sulla selezione della memoria file, fare riferimento alla sezione *5-1-5 Ambiti di applicazione* a pagina 211.

5-1-1 Tipi di memoria file

Categoria	Tipo	Capacità	Modello	Dati del file riconosciuti dalla CPU	Operazioni sui file consentite
Schedine di memoria	Memoria flash	30 MB	HMC-EF372	1) Intero programma utente 2) Intervallo specifico della memoria I/O 3) Dati dell'area dei parametri (configurazione del PLC e altre impostazioni) Vedere nota 4.	Tutte (per ulteriori informazioni, vedere pagina 209).
		64 MB	HMC-EF672		
Memoria file nell'area EM Area EM Banco 0 Banco 1 Banco n Banco C Memoria file nell'area EM	RAM	Capacità dell'area EM delle CPU Serie CS CS1H-CPU67H: 832 KB (banchi da 0 a C: da E0_00000 a EC_00000) Serie CJ CJ1H-CPU66H: 448 KB (banchi da 0 a 6: da E0_00000 a E6_00000)	Dal banco specificato nell'area EM della memoria I/O all'ultimo banco (specificato nella configurazione del PLC)		Non è possibile trasferire dati dalla memoria file nell'area EM con la funzione di trasferimento automatico all'avvio (per ulteriori informazioni, vedere pagina 209).

Nota

1. Per ulteriori informazioni sull'installazione e la rimozione delle schedine di memoria, fare riferimento alla sezione *5-2 Gestione dei file*.
2. Inizializzare la memoria file nell'area EM o la schedina di memoria prima di utilizzarla per la prima volta. Per ulteriori informazioni sull'inizializzazione, fare riferimento alla sezione *5-3 Utilizzo della memoria file*.
3. È possibile utilizzare l'adattatore per schedine di memoria HMC-AP001 per installare una schedina di memoria nello slot per schede PLC di un personal computer e utilizzare la schedina come dispositivo di memorizzazione.
4. Quando si utilizza CX-Programmer, la CPU riconosce le tabelle dei simboli (inclusi i commenti degli I/O) e i commenti. Se è stata installata una schedina di memoria, la destinazione del trasferimento sarà la schedina di memoria. Se non è installata alcuna schedina di memoria, la destinazione sarà la memoria file nell'area EM.

<u>**Precauzioni relative alla schedina di memoria**</u>

Prima di utilizzare una schedina di memoria, prendere in considerazione i seguenti punti.

Formato

Le schedine di memoria sono già formattate al momento dell'acquisto, non è quindi necessario formattarle. Per formattare una schedina dopo l'utilizzo, utilizzare sempre la CPU e CX-Programmer o una Console di programmazione.

Se la schedina di memoria viene formattata direttamente in un notebook o in un altro computer, la CPU potrebbe non riconoscerla. In tal caso, non sarà possibile utilizzare la schedina anche dopo averla riformattata nella CPU.

Numero di file nella directory principale

Il numero di file che è possibile memorizzare nella directory principale di una schedina di memoria è limitato, così come esistono limitazioni per i dischi rigidi. Il limite dipende dal tipo e dal formato della schedina di memoria, ma sarà comunque compreso tra 128 e 512 file. Quando si utilizzano applicazioni che prevedono la scrittura di file di log o altri file a intervalli periodici, impostare il programma per la scrittura dei file in una sottodirectory anziché nella directory principale.

È possibile creare sottodirectory utilizzando un computer o l'istruzione CMND(490). Per un esempio specifico di utilizzo dell'istruzione CMND(490),

fare riferimento a *3-25-5 DELIVER COMMAND: CMND(490)* nel manuale *CS/CJ Series Instructions Reference Manual*.

Numero di operazioni di scrittura

In generale, non esistono limitazioni relative al numero di operazioni di scrittura che è possibile eseguire nella memoria flash. Ai fini della garanzia, tuttavia, per le schedine di memoria è stato stabilito un limite di 100.000 operazioni di scrittura. Ad esempio, se si esegue un'operazione di scrittura sulla schedina di memoria ogni 10 minuti, nell'arco di due anni verranno eseguite più di 100.000 operazioni di scrittura.

Dimensione minima dei file

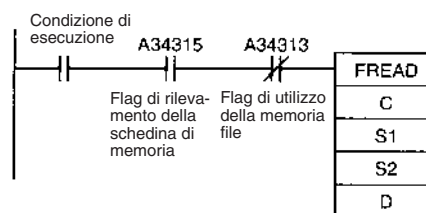
Se nella schedina di memoria vengono memorizzati numerosi file di dimensioni ridotte, ad esempio contenenti solo pochi dati dell'area DM, non sarà possibile sfruttare completamente la capacità della schedina. Se si utilizza una schedina di memoria con un'unità di allocazione di 4.096 byte, per ciascun file verranno occupati almeno 4.096 byte di memoria, indipendentemente dalle dimensioni effettive del file. Se si salvano 10 canali di dati dell'area DM sulla schedina di memoria, verranno utilizzati 4.096 byte di memoria, anche se la dimensione effettiva del file è di 68 byte. L'impiego di file di dimensioni così ridotte riduce notevolmente il fattore di utilizzo della schedina di memoria. Tuttavia, se per migliorare l'utilizzo si riducono le dimensioni dell'unità di allocazione, verrà ridotta anche la velocità di accesso.

È possibile verificare le dimensioni dell'unità di allocazione della schedina di memoria utilizzando il comando CHKDSK da un prompt del DOS. La procedura specifica non è riportata in questa sede. Per ulteriori informazioni sulle dimensioni dell'unità di allocazione, consultare la documentazione del computer.

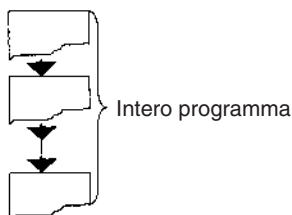
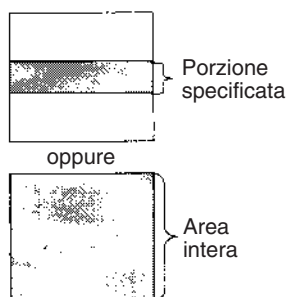
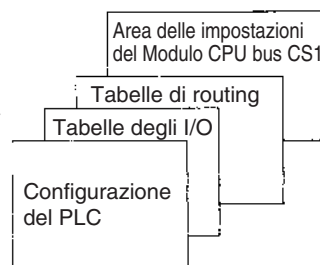
Precauzioni per l'accesso alla schedina di memoria

Quando il PLC accede alla schedina di memoria, l'indicatore BUSY della CPU si illumina. Osservare le seguenti precauzioni.

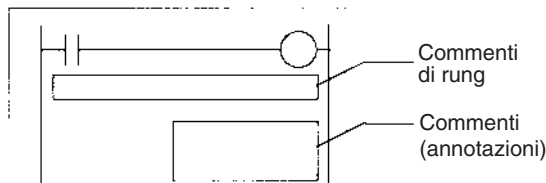
1,2,3...

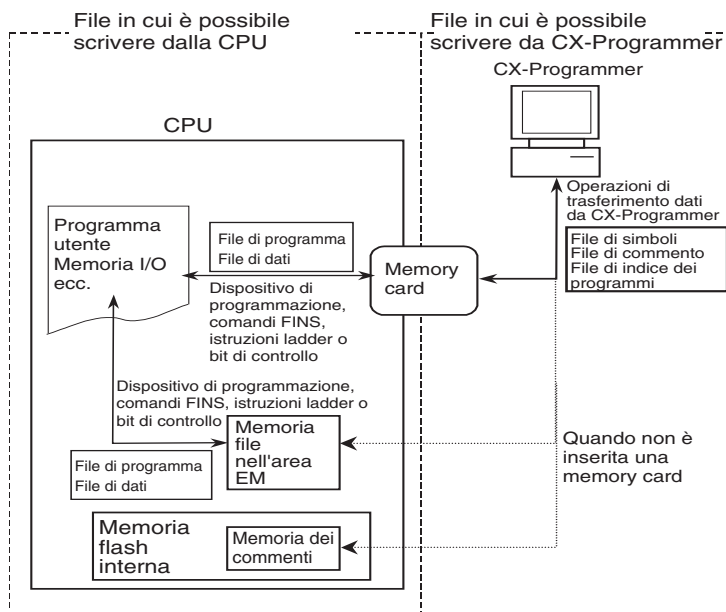
1. Non spegnere mai la CPU quando l'indicatore BUSY è acceso. Ciò può rendere inutilizzabile la schedina di memoria.
2. Non rimuovere mai la schedina di memoria dalla CPU quando l'indicatore BUSY è illuminato. Prima di rimuovere la schedina di memoria, premere il pulsante di alimentazione della schedina per interrompere l'alimentazione e attendere che l'indicatore BUSY si spenga. In caso contrario, la schedina di memoria potrebbe venire danneggiata e non essere più utilizzabile.
3. Inserire la schedina di memoria con l'etichetta rivolta a destra. Non tentare di inserirla con qualsiasi altro orientamento. Ciò potrebbe danneggiare la schedina di memoria o la CPU.
4. Dopo l'inserimento, sono sufficienti pochi secondi per il riconoscimento della schedina di memoria da parte della CPU. Quando si effettua l'accesso a una schedina di memoria subito dopo l'accensione o l'inserimento della schedina, programmare una condizione NC per il flag di riconoscimento della schedina di memoria (A34315) come condizione di ingresso, come indicato di seguito.

5-1-2 Dati contenuti nei file

Nella seguente tabella sono descritti i tipi di file in cui è possibile scrivere..

Tipo di file	Funzionamento
• File di programma • File di dati • File di parametri	Per accedere (lettura, scrittura, ecc.) a questi file è possibile utilizzare un dispositivo di programmazione (CX-Programmer o Console di programmazione), comandi FINS, istruzioni ladder, operazioni di backup semplice o i bit di controllo speciali nella memoria della CPU.
• File delle tabelle dei simboli • File di commento • File di indice dei programmi	Per accedere (lettura, scrittura, ecc.) a questi file è possibile utilizzare CX-Programmer o operazioni di backup semplice.

Dati dell'area dei parametri:
File dei parametri
Impostazioni iniziali della CPU

File di tabelle dei simboli
Tabelle di variabili usate da CX-Programmer

Simboli, indirizzi, tipi di dati, commenti di I/O

File di commenti
Commenti usati da CX-Programmer

File di indice del programma
Informazioni di sezione (utilizzate da CX-Programmer)

Nomi delle sezioni, commenti delle sezioni

Nota Con CX-Programmer è possibile trattare le tabelle dei simboli (simboli, indirizzi e commenti I/O) come file.

File	Nome file	Estensione	Contenuto
File di tabella dei simboli	SYMBOLS	.SYM	Simboli globali e locali
File di commento	COMMENTS	.CMT	Commenti di rung e commenti (annotazioni)
File di indice del programma	PROGRAM	.IDX	Nomi delle sezioni e commenti delle sezioni

È possibile eseguire operazioni di trasferimento dati per i progetti creati con CX-Programmer per trasferire tutti i file sopra menzionati (file delle tabelle dei simboli, file di commento, file di indice dei programmi) dalla CPU a una schedina di memoria e viceversa o tra memorie file nell'area EM. I trasferimenti dei file di indice dei programmi sono supportati a partire dalla versione 2.0. È inoltre possibile trasferire i file delle tabelle dei simboli e i file di commento tra CX-Programmer, RAM del computer e un dispositivo di memorizzazione dei dati utilizzando CX-Programmer 1.2 o una versione successiva.

5-1-3 File

I file vengono formattati in DOS e, di conseguenza, possono essere utilizzati come normali file in un computer che esegue Windows.

I file vengono identificati per mezzo di nomi ed estensioni, come indicato nella tabella che segue. I caratteri che è possibile utilizzare per i nomi dei file sono: lettere da A a Z, numeri da 0 a 9 nonché i caratteri !, &, $, #, `, {, }, –, ^, (,), e _.

Non è possibile utilizzare nei nomi di file i seguenti caratteri: ,, ., /, ¥, ?, *, “, :, :, <, >, =, +, spazio e caratteri a 2 byte.

L'estensione del file dipende dal tipo di file memorizzato. I file di dati hanno estensione IOM, TXT, CSV o IOR. Le estensioni TXT, CSV e IOR non sono supportati dalle CPU CS1 serie CS precedenti a EV1. Ai file di programma viene associata l'estensione OBJ, mentre ai file dei parametri viene assegnata l'estensione STD. È possibile specificare la posizione di un file in memoria nella directory e una directory può contenere fino a 5 livelli di sottodirectory (compresa la directory principale).

Tipi, nomi ed estensioni dei file

Tramite la CPU è possibile gestire (leggere e scrivere) tre tipi di file.

- **File di uso generico**
  È possibile accedere in lettura o scrittura a questi file mediante dispositivi di programmazione, comandi FINS, istruzioni o operazioni dei bit di controllo dell'area ausiliaria. L'utente può definire liberamente i nomi dei file.
- **File per il trasferimento automatico all'avvio**
  Questi file vengono trasferiti automaticamente dalla schedina di memoria alla CPU quando si accende il sistema. Il nome del file è AUTOEXEC o ATEXEC□□ quando vengono trasferiti i file dei parametri. Il nome del file è REPLACE quando non vengono trasferiti i file dei parametri (solo CPU della serie CS/CJ versione 2.0 o successiva).
- **File di backup** (non supportati dalle CPU CS1 serie CS precedenti a EV1)
  Questi file vengono trasferiti dalla schedina di memoria alla CPU e viceversa mediante la funzione di backup. I nomi dei file sono fissi e corrispondono a BACKUP□□.

  Nota Utilizzare la memory card, la memoria file nell'area EM o la memoria dei commenti per i file delle tabelle dei simboli, i file di commento e i file di indice dei programmi.
- **File di sistema per CX-Programmer**
  Questi file vengono generati automaticamente quando si esegue il trasferimento tramite CX-Programmer. I nomi dei file sono fissi.

File di uso generico

Nella seguente tabella sono indicati i nomi e le estensioni dei file di uso generico.

<table>
<tr><th>Tipo</th><th>Nome[1]</th><th>Estensione</th><th>Descrizione</th><th colspan="2">Spiegazione</th></tr>
<tr><td>File di programma</td><td>********</td><td>.OBJ</td><td>Intero programma utente</td><td colspan="2">• Tutti i task ciclici e ad interrupt, nonché i dati dei task per una CPU.</td></tr>
<tr><td>File dell'area dei parametri</td><td>********</td><td>.STD</td><td>Configurazione del PLC, tabella di I/O registrati, tabelle di routing, impostazioni del Modulo CPU bus[3], ecc.</td><td colspan="2">• Include tutte le impostazioni iniziali per una CPU.
• Non occorre suddividere i dati di parametro per tipo all'interno del file.</td></tr>
<tr><td rowspan="3">File di dati</td><td rowspan="3">********</td><td>.IOM</td><td rowspan="3">Intervallo specificato nella memoria I/O</td><td rowspan="3">• Dati presenti in unità canale (16 bit), dal canale di inizio a quello di fine, situati in un'unica area.
• L'area può essere l'area CIO, HR, WR, AR, DM o EM.</td><td>Formato binario</td></tr>
<tr><td>.TXT</td><td>Formato TXT[2] (non delimitato o delimitato da tabulazioni)</td></tr>
<tr><td>.CSV</td><td>Formato CSV[2] (delimitato da virgole)</td></tr>
</table>

Nota
1. I nomi di file rappresentati nella tabella mediante "********" sono costituiti da un massimo di 8 caratteri ASCII.
2. I formati di file TXT e CSV non sono supportati dalle CPU CS1 serie CS precedenti a EV1.
3. Un esempio di impostazioni del Modulo CPU bus è dato dalle tabelle data link. Per informazioni su altri dati di configurazione, vedere il manuale dell'operatore del modulo specifico.

File trasferiti automaticamente all'avvio

Nella colonna *File* sono indicati i file che devono essere presenti nella schedina di memoria per consentire il trasferimento automatico all'avvio.

Sono disponibili due metodi per il trasferimento automatico dei file all'avvio: trasferimento con un file dell'area dei parametri e trasferimento senza un file dell'area dei parametri.

Trasferimento con un file dell'area dei parametri

Tipo	Nome[1]	Estensione	Descrizione	Spiegazione	File
File di programma	AUTOEXEC	.OBJ	Intero programma utente	• Anche quando si specifica il trasferimento automatico all'avvio, non è necessario che questo file si trovi nella schedina di memoria. • Contiene tutti i programmi dei task ciclici e ad interrupt nonché i dati dei task per una CPU. • Il trasferimento è consentito solo se la schedina di memoria contiene anche un file dell'area dei parametri (AUTOEXEC.STD).	Richiesta
File dell'area dei parametri	AUTOEXEC	.STD	Configurazione del PLC, tabella di I/O registrati, tabelle di routing, impostazioni del Modulo CPU bus[3], ecc.	Quando si specifica il trasferimento automatico all'avvio, questo file deve essere presente nella schedina di memoria. Include tutte le impostazioni iniziali per una CPU. Non occorre suddividere i dati di parametro per tipo all'interno del file. I dati di impostazione iniziale vengono automaticamente memorizzati all'avvio in posizioni specifiche della CPU. Se la schedina di memoria contiene un file di programma denominato REPLACE.OBJ, il file dell'area dei parametri non verrà trasferito.	Richiesta
File di dati	AUTOEXEC	.IOM	Dati della memoria I/O (contiene il numero specificato di canali di dati a partire da D20000)	• I dati DM a partire dall'indirizzo D20000 vengono memorizzati in un file denominato AUTOEXEC.IOM. • All'avvio, tutti i dati del file vengono trasferiti all'area DM a partire dall'indirizzo D20000. • Quando si utilizza la funzione di trasferimento automatico all'avvio, non è necessario che questo file si trovi nella schedina di memoria.	---
	ATEXECDM	.IOM	Dati della memoria I/O[2] (contiene il numero specificato di canali di dati a partire da D00000)	• I dati DM a partire dall'indirizzo D00000 vengono memorizzati in un file denominato ATEXECDM.IOM. • All'avvio, tutti i dati del file vengono trasferiti all'area DM a partire dall'indirizzo D00000. • Quando si utilizza la funzione di trasferimento automatico all'avvio, non è necessario che questo file si trovi nella schedina di memoria. **Nota** I dati di questo file hanno la priorità in caso di sovrapposizione con i dati DM contenuti in AUTOEXEC.IOM.	---
	ATEXECE□	.IOM	Dati dell'area EM (banco □)[2] (contiene il numero specificato di canali di dati a partire da E□_00000)	• I dati per il banco EM □ a partire da E□_00000 vengono memorizzati in un file denominato ATEXECE□.IOM. Il numero massimo di banchi dipende dal modello di CPU in uso. • All'avvio, tutti i dati del file vengono trasferiti al banco EM □ a partire da E□_00000. • Quando si utilizza la funzione di trasferimento automatico all'avvio, non è necessario che questo file si trovi nella schedina di memoria.	---

Trasferimento senza un file dell'area dei parametri

Tipo	Nome[1]	Estensione	Descrizione	Spiegazione	File
File di programma	REPLACE Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	.OBJ	Intero programma utente	• Il contenuto è identico a quello del file AUTOEXEC.OBJ. • Il file verrà trasferito all'avvio, anche se non è disponibile alcun file dell'area dei parametri (AUTOEXEC.STD).	Richiesta
File dell'area dei parametri	Non richiesta	---	---	Il file dell'area dei parametri non verrà trasferito, a prescindere dal nome del file.	---
File di dati	REPLACE Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	.IOM	Dati della memoria I/O (contiene il numero specificato di canali di dati a partire da D20000)	• Il contenuto è identico a quello del file AUTOEXEC.IOM. • Questo file verrà trasferito all'avvio se la schedina di memoria contiene anche un file di programma denominato REPLACE.OBJ.	---
	REPLCDM Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	.IOM	Dati della memoria I/O (contiene il numero specificato di canali di dati a partire da D00000).	• Il contenuto è identico a quello del file ATEXECDM.IOM. • Questo file verrà trasferito all'avvio se la schedina di memoria contiene anche un file di programma denominato REPLACE.OBJ.	---
	REPLCE□ Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	.IOM	Dati dell'area EM (banco □) (contiene il numero specificato di canali di dati a partire da E□_00000).	• Il contenuto è identico a quello del file ATEXECE□.IOM. • Questo file verrà trasferito all'avvio se la schedina di memoria contiene anche un file di programma denominato REPLACE.OBJ.	---

Nota

1. Verificare che i nomi dei file da trasferire automaticamente all'avvio siano AUTOEXEC o ATEXEC□□.
2. I file ATEXECDM.IOM e ATEXECE□.IOM non sono supportati dalle CPU CS1 serie CS precedenti a EV1.
3. Un esempio di impostazioni del Modulo CPU bus è dato dalle tabelle data link. Per informazioni su altri dati di configurazione, vedere il manuale dell'operatore del modulo specifico.

File di backup (non supportati dalle CPU CS1 serie CS precedenti a EV1)

I file riportati nella tabella che segue vengono creati in modo automatico quando i dati vengono trasferiti da e verso la schedina di memoria durante le operazioni di backup.

Tipo	Nome[1]	Estensione	Descrizione	Spiegazione
File di dati	BACKUP	.IOM	Canali dell'area DM allocati a Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS)	• Contiene i dati DM nell'intervallo si indirizzi da D20000 a D32767. • Questo file deve essere presente nella schedina di memoria quando i dati vengono letti dalla schedina durante il backup.
	BACKUPIO	.IOR	Aree di dati della memoria I/O	• Contiene tutti i dati delle aree CIO, WR, HR e AR, nonché i flag di completamento e i valori attuali di temporizzatori e contatori.[2] • Questo file deve essere presente nella schedina di memoria quando i dati vengono letti dalla schedina durante il backup.
	BACKUPDM	.IOM	Area DM di uso generico	• Contiene i dati DM nell'intervallo di indirizzi da D00000 a D19999. • Questo file deve essere presente nella schedina di memoria quando i dati vengono letti dalla schedina durante il backup.
	BACKUPE□	.IOM	Area EM di uso generico	Contiene tutti i dati EM per il banco EM □ con indirizzi compresi tra E□_00000 e E□_32767 (il numero massimo di banchi dipende dal modello di CPU utilizzato). Questo file deve essere presente nella schedina di memoria quando i dati vengono letti dalla schedina durante il backup. • Quando si esegue il backup dei dati sulla schedina di memoria, i dati di ciascun banco EM vengono automaticamente scritti in un file distinto.
File di programma	BACKUP	.OBJ	Intero programma utente	• Contiene tutti i programmi dei task ciclici e ad interrupt nonché i dati dei task per una CPU. • Questo file deve essere presente nella schedina di memoria quando i dati vengono letti dalla schedina durante il backup.
File dei parametri		.STD	Configurazione del PLC, tabella di I/O registrati, tabelle di routing, impostazioni del Modulo CPU bus[3], ecc.	• Contiene tutte le impostazioni iniziali per una CPU. • Non occorre suddividere i dati di parametro per tipo all'interno del file. • Questo file deve essere presente nella schedina di memoria quando i dati vengono letti dalla schedina durante il backup.
File di backup di moduli e schede specifici (solo CPU CS1-H, CJ1-H o CJ1M)	BACKUP□□ (dove □□ è l'indirizzo del modulo o della scheda di cui si esegue il backup)	.PRM	Dati relativi a schede o moduli specifici	• Dati di backup di controllo di una scheda o modulo. Per ulteriori informazioni, fare riferimento alla sezione *5-2-6 Funzione di backup semplice*.
File delle tabelle dei simboli (vedere nota 1)	BKUPSYM	.SYM	Tabelle dei simboli globali, tabelle dei simboli locali e dati di configurazione dell'area allocati automaticamente	Sono inclusi i seguenti dati all'interno delle tabelle dei simboli globali/locali di CX-Programmer: variabili, indirizzi, tipi di dati, commenti degli I/O È incluso l'insieme di dati nelle impostazioni di allocazione automatica degli indirizzi del PLC di CX-Programmer.
File di commento (vedere nota 1)	BKUPCMT	.CMT	Commenti di rung e commenti	Commenti di rung e commenti di CX-Programmer.
File di indice dei programmi (vedere nota 1)	BKUPPRG	.IDX	Nomi e commenti di sezione e commenti di programma di CX-Programmer	I dati che delimitano le sezioni di CX-Programmer (la posizione del delimitatore dipende però dal delimitatore di sezione all'interno del programma).

Nota 1. I seguenti file di backup possono essere creati solo se si utilizzano CPU della serie CS/CJ versione 3.0 o successiva.
File delle tabelle dei simboli, file di commento e file di indice dei programmi
Questi file vengono creati automaticamente dai file nella memory card, nella memoria file nell'area EM o nella memoria dei commenti.
2. Un esempio di impostazioni del Modulo CPU bus è dato dalle tabelle data link. Per informazioni su altri dati di configurazione, vedere il manuale dell'operatore del modulo specifico.

File di sistema per CX-Programmer

Questi file vengono in genere creati automaticamente quando vengono scaricati dei dati tramite CX-Programmer versione 5.0 o successiva. I nomi dei file sono fissi. Se si utilizza CX-Programmer versione 5.0 o successiva con CPU versione 3.0 o successiva, durante il trasferimento di progetti è possibile selezionare una delle seguenti opzioni di memoria come destinazione di trasferimento per i file di sistema.

- Memory card
- Memoria file nell'area EM
- Memoria dei commenti (nella memoria flash della CPU)

Nota Se si utilizza CX-Programmer versione 4.0 o precedente, non è possibile memorizzare questi file nella memoria dei commenti, anche se si utilizza una CPU versione 3.0 o successiva.

Tipo	Nome	Estensione	Descrizione	Spiegazione	
File delle tabelle dei simboli	SYMBOLS	.SYM	Tabelle dei simboli globali e locali	Questi file vengono creati automaticamente quando vengono scaricati dei dati tramite CX-Programmer.	Sono inclusi i seguenti dati all'interno delle tabelle dei simboli globali/locali di CX-Programmer: variabili, indirizzi, tipi di dati, commenti degli I/O È incluso l'insieme di dati nelle impostazioni di allocazione automatica degli indirizzi del PLC di CX-Programmer.
File di commento	COMMENTS	.CMT	Commenti di rung e commenti (annotazioni)		Commenti di rung e commenti di CX-Programmer.
File di indice dei programmi	PROGRAM	.IDX	Nomi delle sezioni, commenti delle sezioni		I dati che delimitano le sezioni di CX-Programmer (la posizione del delimitatore dipende però dal delimitatore di sezione all'interno del programma). Nota: Solo CX-Programmer versione 2.0 o successiva

Nota Se si utilizza CX-Programmer versione 1.2 o successiva, è possibile trasferire in linea i file delle tabelle dei simboli e i file di commento descritti nella precedente tabella tra CX-Programmer e la RAM del PC e tra la RAM del PC e un dispositivo di memorizzazione di massa.

Directory

Con i PLC della serie CS/CJ è possibile accedere ai file contenuti nelle sottodirectory, mentre le Console di programmazione consentono di accedere solo ai file presenti nella directory principale. La lunghezza massima di un percorso di directory è di 65 caratteri. Quando si creano sottodirectory nella schedina di memoria utilizzando Windows, accertarsi di non superare il numero massimo di caratteri.

Dimensioni dei file

È possibile calcolare le dimensioni dei file in byte utilizzando le equazioni riportate nella seguente tabella:

Tipo di file	Dimensioni del file
File di dati (.IOM)	(Numero di canali × 2) + 48 byte Esempio intera area DM (da D00000 a D32767) (32.768 canali × 2) + 48 = 65.584 byte
File di dati (.TXT o .CSV)	Le dimensioni del file dipendono dal numero di delimitatori e di ritorni a capo utilizzati. Il codice di delimitatore occupa un byte, mentre il codice di ritorno a capo impegna due byte. Esempio 1: canali non delimitati, nessun ritorno a capo 123456789ABCDEF012345678 occupa 24 byte. Esempio 2: canali delimitati, ritorno a capo ogni 2 campi 1234,5678↵ 9ABC,DEF0↵ 1234,5678↵ occupa 33 byte. Esempio 3: canali doppi delimitati, ritorno a capo ogni 2 campi 56781234,DEF01234↵ 56781234↵ occupa 29 byte.
File di programma (.OBJ)	(Numero di step × 4) + 48 byte (vedere nota)
File dei parametri (.STD)	16.048 byte

Nota Calcolare il numero di step nel file di programma sottraendo gli step UM disponibili dagli step UM totali. Tali valori sono indicati nel rapporto dei riferimenti incrociati di CX-Programmer. Per ulteriori informazioni, fare riferimento al manuale *CX-Programmer Operation Manual*.

File di dati

File di uso generico

1,2,3... 1. I file di dati di uso generico hanno estensione IOM, TXT o CSV. I formati di file TXT e CSV non sono supportati dalle CPU CS1 serie CS precedenti a EV1.

Estensione	Formato dei dati	Contenuto		Canali per campo
.IOM	Binario	Formato dei dati per la serie CS/CJ		---
.TXT (vedere note)	Canali non delimitati	Formato ASCII	Questo formato viene creato convertendo i campi a un canale della memoria I/O (esadecimale a 4 cifre) in ASCII e comprimendo i campi senza delimitatori. Per delimitare i record, è possibile utilizzare i ritorni a capo.	1 canale
	Canali doppi non delimitati		Questo formato viene creato convertendo i campi a due canali della memoria I/O (esadecimale a 8 cifre) in ASCII e comprimendo i campi senza delimitatori. Per delimitare i record, è possibile utilizzare i ritorni a capo.	2 canali
	Canali delimitati da tabulazioni		Questo formato viene creato convertendo i campi a un canale della memoria I/O (esadecimale a 4 cifre) in ASCII e delimitando i campi mediante tabulazioni. Per delimitare i record, è possibile utilizzare i ritorni a capo.	1 canale
	Canali doppi delimitati da tabulazioni		Questo formato viene creato convertendo i campi a due canali della memoria I/O (esadecimale a 8 cifre) in ASCII e delimitando i campi mediante tabulazioni. Per delimitare i record, è possibile utilizzare i ritorni a capo.	2 canali
.CSV (vedere note)	Canali delimitati da virgole		Questo formato viene creato convertendo i campi a un canale della memoria I/O (esadecimale a 4 cifre) in ASCII e delimitando i campi mediante virgole. Per delimitare i record, è possibile utilizzare i ritorni a capo.	1 canale
	Canali doppi delimitati da virgole		Questo formato viene creato convertendo i campi a due canali della memoria I/O (esadecimale a 8 cifre) in ASCII e delimitando i campi mediante virgole. Per delimitare i record, è possibile utilizzare i ritorni a capo.	2 canali

Nota a) Lettura e scrittura di file di dati TXT e CSV:
è possibile leggere e scrivere file di dati TXT e CSV utilizzando solo le istruzioni FREAD(700) e FWRIT(701).

b) Considerazioni sui caratteri:
se il file TXT o CSV contiene caratteri non esadecimali (ovvero diversi dai caratteri compresi tra 0 e 9, A e F o a e f), i dati non verranno scritti in modo corretto nella memoria I/O.

c) Considerazioni sulle dimensioni del campo:
quando si utilizzano i canali, i dati non verranno scritti in modo corretto nella memoria I/O se il file TXT o CSV contiene campi di dimensioni diverse da un valore esadecimale a 4 cifre. Allo stesso modo, quando si utilizzano canali doppi, non è possibile scrivere i dati nella memoria I/O in modo corretto se il file contiene campi di dimensioni diverse da un valore esadecimale a 8 cifre.

d) Ordine di memorizzazione:
quando si utilizzano i canali, i dati della memoria I/O vengono convertiti in ASCII e memorizzati in campi a un canale in ordine crescente, dall'indirizzo di memoria I/O più basso al più alto.
quando si utilizzano canali doppi, i dati della memoria I/O vengono convertiti in ASCII e memorizzati in campi a due canali in ordine crescente, dall'indirizzo di memoria I/O più basso al più alto. All'interno dei campi a due canali, il canale con valore di indirizzo più alto viene memorizzato per primo e il canale con valore di indirizzo più basso per secondo.

e) Delimitatori:
quando non vi sono delimitatori, i campi vengono compressi in sequenza e memorizzati. Se si utilizzano virgole quali delimitatori, prima della memorizzazione vengono inserite delle virgole tra i campi. Se si utilizzano tabulazioni quali delimitatori, prima della memorizzazione vengono inseriti dei codici di tabulazione tra i campi.
Quando si specificano i delimitatori in FREAD(700), siano essi virgole o tabulazioni, i dati vengono letti come delimitati, con delimitatori (virgole o tabulazioni) per singolo canale.

f) Ritorni a capo:
quando non si utilizzano ritorni a capo, i dati vengono compressi in sequenza.
L'utilizzo dei ritorni a capo implica l'inserimento di un ritorno a capo dopo un numero specificato di campi. Se nel file si utilizzano ritorni a capo, non sarà possibile specificare un offset dall'inizio del file (canale di lettura iniziale o canale di scrittura iniziale) nelle istruzioni FREAD(700) e FWRIT(701).

g) Numero di campi:
la quantità complessiva di dati del file dipende dal numero di campi (numero di elementi di scrittura) specificato nell'istruzione FWRIT(701) e dal numero di canali per campo. Quando si utilizzano i canali, si ottengono campi a canale singolo. Viceversa, quando si utilizzano canali doppi, si ottengono campi a canale doppio.

2. I file di dati non contengono informazioni relative al tipo di dati memorizzati, ad esempio l'area di memoria salvata. Per agevolarne la gestione, denominare i file in modo che il nome ne indichi il contenuto, come negli esempi forniti di seguito.

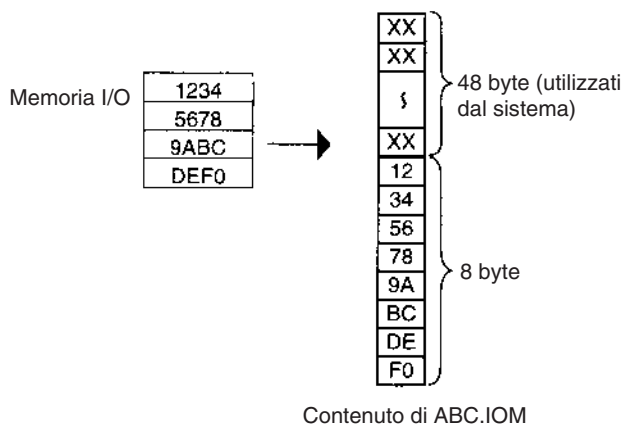
   Esempi: D00100.IOM, CIO0020.IOM

   I dati all'inizio del file verranno scritti a partire dall'indirizzo specificato nella memoria I/O, anche se i dati scritti in origine nel file (IOM, TXT o CSV) non provengono dalla medesima area. Ad esempio, se i dati CIO all'interno di un file vengono scritti nell'area DM da un dispositivo di programmazione, i dati verranno letti nell'area DM della CPU senza alcuna indicazione sulla diversità dell'area.

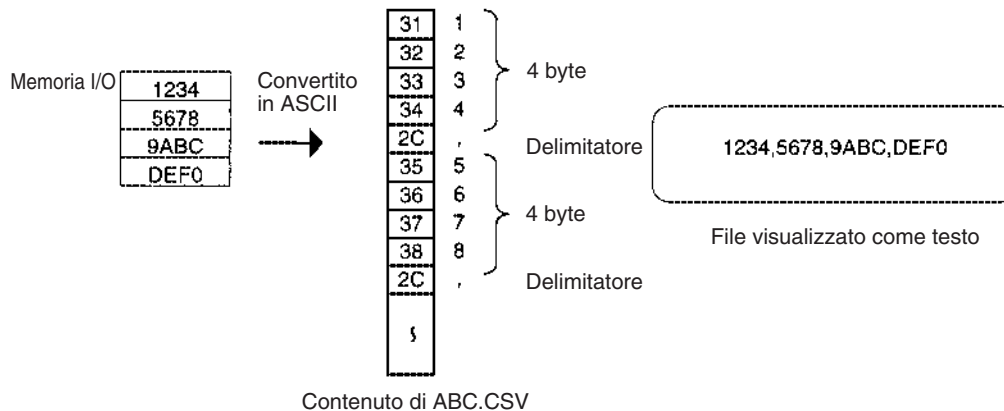
Nota I file di dati in formato TXT e CSV contengono dati esadecimali (da 0 a 9, da A a F) che consentono di utilizzare i dati numerici della memoria I/O in fogli elettronici.

Struttura dei file di dati IOM

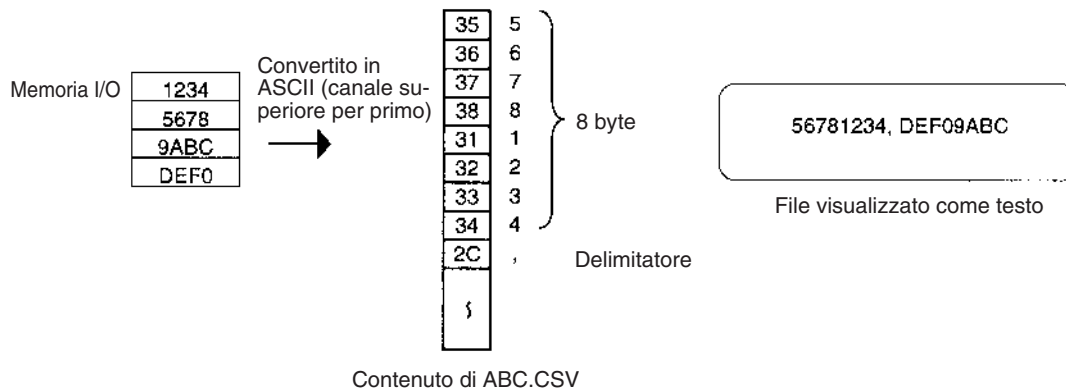
Nella figura seguente è illustrata la struttura dei dati binari di un file di dati (ABC.IOM) contenente quattro canali della memoria I/O, ovvero i valori esadecimali 1234, 5678, 9ABC e DEF0. L'utente tuttavia non deve considerare il formato dei dati nell'utilizzo normale.

Contenuto di ABC.IOM

Struttura dei file di dati CSV e TXT (canale singolo)

Nella figura seguente è illustrata la struttura dei dati di un file di dati CSV (ABC.CSV) con campi a canale singolo contenente quattro canali della memoria I/O, ovvero i valori esadecimali 1234, 5678, 9ABC e DEF0. La struttura di un file TXT con campi a canale singolo è identica.

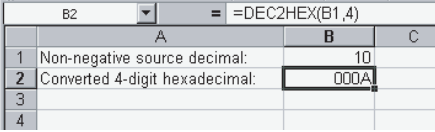
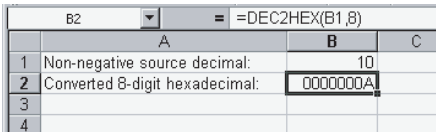
Contenuto di ABC.CSV

Struttura dei file di dati CSV e TXT (canale doppio)

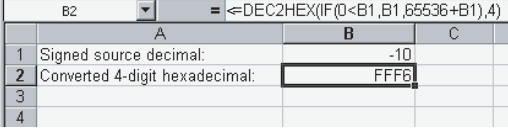
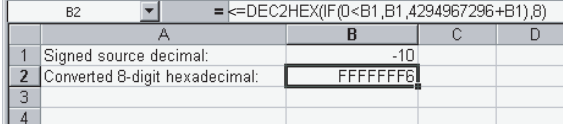
Nella figura seguente è illustrata la struttura dei dati di un file di dati CSV (ABC.CSV) con campi a canale doppio contenente quattro canali della memoria I/O, ovvero i valori esadecimali 1234, 5678, 9ABC e DEF0. La struttura di un file TXT con campi a canale doppio è identica.

Contenuto di ABC.CSV

Creazione di file di dati con un foglio elettronico

Per creare file di dati TXT e CSV con un foglio elettronico, quale Microsoft Excel, attenersi alla seguente procedura.

- Impostare il formato delle celle per il contenuto di testo.
- Se si utilizzano campi a canale singolo, digitare 4 caratteri in ciascuna cella. Se si utilizzano campi a canale doppio, digitare 8 caratteri in ciascuna cella. Ad esempio, se si utilizzano campi a canale singolo, digitare 000A e non solo A.
- Immettere nelle celle solo caratteri esadecimali (da 0 a 9, da A a F o da a a f). Non è possibile utilizzare altri caratteri e codici.

Quando si desidera memorizzare nella memoria I/O le cifre esadecimali, è utile convertire in esadecimali i valori decimali inseriti del foglio elettronico. Per convertire i valori decimali in esadecimali, seguire la procedura descritta.

1,2,3...

1. Scegliere ***Componenti aggiuntivi...*** dal menu Strumenti.
2. Selezionare **Strumenti di analisi** dalla finestra di dialogo Componenti aggiuntivi.
3. Scegliere ***Funzione*** dal menu Inserisci dopo avere selezionato la cella in cui si desidera utilizzare la funzione.

4. Selezionare la funzione ***DECIMALE.HEX (numero, cifre)*** dalla categoria Ingegneristiche.
5. Per la conversione in esadecimale a 4 cifre, immettere quanto segue per la variabile del numero: IF(0<=posizione cella,posizione cella,65535+posizione cella)

   Per la conversione in esadecimale a 8 cifre, immettere quanto segue per la variabile del numero: IF(0<=posizione cella,posizione cella,4294967296+posizione cella)

- **Esempio 1:** immissione di valori decimali non negativi.

Elemento	Conversione di decimali senza segno in esadecimali a 4 cifre	Conversione di decimali senza segno in esadecimali a 8 cifre
Funzione utilizzata	DECIMALE.HEX(*posizione_cella*0,4)	DECIMALE.HEX(*posizione_cella*,8)
Esempio	Immissione di 10 in formato decimale e conversione in 000A in formato esadecimale a 4 cifre. B2 = =DEC2HEX(B1,4) 1 Non-negative source decimal: 10 2 Converted 4-digit hexadecimal: 000A	Immissione di 10 in formato decimale e conversione in 0000000A in formato esadecimale a 8 cifre. B2 = =DEC2HEX(B1,8) 1 Non-negative source decimal: 10 2 Converted 8-digit hexadecimal: 0000000A

- **Esempio 2:** immissione di valori decimali con segno.

Elemento	Conversione di decimali con segno in esadecimali a 4 cifre	Conversione di decimali con segno in esadecimali a 8 cifre
Funzione utilizzata	DECIMALE.HEX(IF(0<=*posizione_cella*,*posizione_cella*,65536+*posizione_cella*),4)	DECIMALE.HEX(IF(0<=*posizione_cella*,*posizione_cella*,4294967296+*posizione_cella*),8)
Esempio	Immissione di -10 in formato decimale e conversione in FFF6 in formato esadecimale a 4 cifre. B2 = =DEC2HEX(IF(0<B1,B1,65536+B1),4) 1 Signed source decimal: -10 2 Converted 4-digit hexadecimal: FFF6	Immissione di -10 in formato decimale e conversione in FFFFFFF6 in formato esadecimale a 8 cifre. B2 = =DEC2HEX(IF(0<B1,B1,4294967296+B1),8) 1 Signed source decimal: -10 2 Converted 8-digit hexadecimal: FFFFFFF6

File di dati trasferiti automaticamente all'avvio

Esistono tre tipi di file che vengono trasferiti automaticamente all'avvio quando viene utilizzata la funzione corrispondente.

- AUTOEXEC.IOM o REPLACE.IOM: canali DM assegnati a Moduli di I/O speciale e a schede interne.
  Il contenuto di questo file viene trasferito all'area DM a partire da D20000 all'accensione del sistema.
- ATEXECDM.IOM o REPLCDM.IOM: canali DM di uso generico.
  Il contenuto di questo file viene trasferito all'area DM a partire dall'indirizzo D00000 all'accensione del sistema.
- ATEXECE□.IOM o REPLCE□.IOM: canali EM di uso generico.
  Il contenuto di questo file viene trasferito all'area EM a partire da E□_00000 all'accensione del sistema.

Quando si creano i file di dati sopra citati, specificare sempre il primo indirizzo come indicato (D20000, D00000 o E□_00000) e accertarsi che le dimensioni del file non superino la capacità dell'area dati specificata.

Tutti i dati in ciascun file verranno sempre trasferiti a partire dal primo indirizzo specificato (D20000, D00000 o E□_00000).

Nota 1. Quando si creano i file AUTOEXEC.IOM, ATEXECDM.IOM e ATEXECE□.IOM o i file REPLACE.IOM, REPLCDM.IOM o

REPLCE□.IOM utilizzando un dispositivo di programmazione (Console di programmazione o CX-Programmer), specificare sempre il primo indirizzo appropriato (D20000, D00000 o E□_00000) e accertarsi che le dimensioni del file non superino la capacità dell'area DM o del banco EM specificato. Il contenuto del file verrà sempre trasferito a partire dal primo indirizzo appropriato a seconda dell'area dati (D20000, D00000 o E□_00000) anche se si specifica un canale iniziale diverso, nel qual caso il contenuto di tale parte dell'area DM o del banco EM verrà sovrascritto da dati errati. Inoltre, se si supera la capacità dell'area DM o del banco EM, come può accadere quando si eseguono le impostazioni da CX-Programmer, i dati rimanenti verranno scritti nel banco EM 0, se l'area DM è piena, oppure nel banco EM successivo qualora sia stata superata la capacità del banco EM specificato.

2. Quando si utilizza CX-Programmer, è possibile specificare un file di dati che superi l'indirizzo massimo dell'area DM, ovvero D32767, o l'indirizzo massimo dell'area EM, ossia E□_32767. Se il file AUTOEXEC.IOM supera il limite dell'area DM, i dati rimanenti verranno scritti nell'area EM a partire da E0_00000, proseguendo secondo l'ordine di indirizzi di memoria e banchi fino all'ultimo banco. È pertanto possibile trasferire automaticamente i dati a entrambe le aree DM ed EM all'avvio. Analogamente, se le dimensioni del file ATEXECE□.IOM superano la capacità del banco EM, i dati rimanenti verranno scritti nei banchi EM successivi.
3. È possibile modificare le configurazioni di sistema per Moduli di I/O speciale, Moduli CPU bus e la scheda interna (solo per la serie CS) utilizzando file AUTOEXEC.IOM distinti che contengono impostazioni diverse per l'area del Modulo di I/O speciale (da D20000 a D29599), l'area del Modulo CPU bus (da D30000 a D31599) e l'area della scheda interna (solo serie CS, da D32000 a D32099). È quindi possibile utilizzare le schedine di memoria per creare librerie di dati di configurazione di sistema per Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS) per sistemi o dispositivi diversi.

File di dati di backup

La funzione di backup consente di creare quattro tipi di file di dati, come descritto di seguito.

Per eseguire il backup dei dati, impostare il pin 7 del DIP switch della CPU su ON e il pin 8 su OFF, inserire la schedina di memoria, quindi tenere premuto l'interruttore di alimentazione della schedina per tre secondi. I quattro file di backup (BACKUP.IOM, BACKUPIO.IOR, BACKUPDM.IOM e BACKUPE□.IOM) verranno creati automaticamente e salvati sulla schedina di memoria.

I file di backup vengono utilizzati esclusivamente dalla funzione di backup, sebbene sia possibile creare tre di questi file, ovvero BACKUP.IOM, BACKUPDM.IOM e BACKUPE□.IOM, mediante le funzioni del dispositivo di programmazione. Il file BACKUPIO.IOR, invece, non può essere creato utilizzando il dispositivo di programmazione.

5-1-4 Descrizione delle procedure operative per i file

Nella tabella che segue sono sintetizzati i sei metodi che è possibile utilizzare per leggere e scrivere file.

Lettura: questa operazione trasferisce i file dalla memoria file alla CPU.
Scrittura: questa operazione trasferisce i file dalla CPU alla memoria file.

OK: possibile; ---: non possibile

Procedura operativa		**Tipo di memoria**	**Nome file**	**Funzione**	**Intero programma**	**Dati nell'area dati (vedere nota 3)**	**Dati nell'area parametri**	**File delle tabelle dei simboli, file di commento e file di indice dei programmi (vedere nota 6)**
Dispositivo di programmazione	CX-Programmer o Console di programmazione	Memory card Memoria file nell'area EM	Qualsiasi nome di file valido	Lettura	**OK**	**OK**	**OK**	---
				Scrittura	**OK**	**OK**	**OK**	---
				Altre operazioni (vedere nota 2)	**OK**	**OK**	**OK**	---
Dispositivo di programmazione	Solo CX-Programmer	Memory card Memoria file nell'area EM, memoria dei commenti	File di sistema per CX-Programmer (nomi di file fissi)	Lettura	---	---	---	**OK**
				Scrittura	---	---	---	**OK** (vedere nota 6)
				Altre operazioni (vedere nota 2)	---	---	---	---
Comando FINS (vedere nota 1)		Memory card Memoria file nell'area EM	Qualsiasi nome di file valido	Lettura	**OK**	**OK**	**OK**	---
				Scrittura	**OK**	**OK**	**OK**	---
				Altre operazioni (vedere nota 2)	**OK** (vedere nota 4)	**OK**	**OK**	---
Istruzioni FREAD(700) e FWRIT(701)		Memory card Memoria file nell'area EM	Qualsiasi nome di file valido	Lettura di dati da un solo file	---	**OK**	---	---
				Scrittura di dati in un solo file	---	**OK**	---	---
Sostituzione dell'intero programma mediante bit di controllo dell'area ausiliaria durante il funzionamento (non supportata dalle CPU CS1 della serie CS precedenti a EV1)		Memory card	Qualsiasi nome di file valido	Lettura	**OK**	---	---	---
Trasferimento automatico all'avvio		Memory card	File di trasferimento automatico all'avvio (AUTOEXEC, ATEXEC□□ o REPLACE (vedere nota 5)	Lettura	**OK**	**OK**	**OK**	---
				Scrittura	---	---	---	---
Operazione di backup (non supportata dalle CPU CS1 della serie CS precedenti a EV1)		Memory card	File di backup semplice (BACKUP□□)	Lettura	**OK**	**OK**	**OK**	**OK**
				Scrittura	**OK**	**OK**	**OK**	**OK** (vedere nota 6)

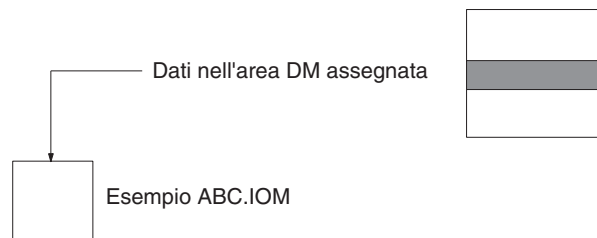
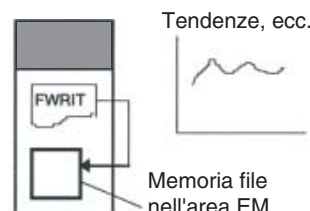
Nota 1. I comandi FINS per le operazioni relative alla memoria file possono essere inviati da computer host collegati tramite Host Link, da un altro PLC connesso a una rete utilizzando l'istruzione CMND(490) o dal programma del PLC locale utilizzando l'istruzione CMND(490). Per quel che concerne le CPU CS1 della serie CS precedenti a EV1, non è possibile eseguire operazioni relative alla memoria file utilizzando CMND(490) nella medesima CPU per cui si stanno eseguendo operazioni di questo tipo.

2. Le altre operazioni includono la formattazione della memoria file, la lettura di dati dei file, la scrittura di dati dei file, la modifica del nome file, la lettura dei dati di memoria dei file, l'eliminazione di file, la copia di file e la creazione di sottodirectory.
3. È possibile leggere e scrivere file di dati in formato TXT e CSV solo mediante le istruzioni FREAD(700) e FWRIT(701). Non è possibile eseguire operazioni di lettura e scrittura su tali file con un dispositivo di programmazione.
4. Per trasferire i file di programma (.OBJ) dalla RAM del computer a un dispositivo di memorizzazione e viceversa, è possibile utilizzare CX-Programmer versione 1.2 e successive.
5. Con le CPU della serie CS/CJ versione 2.0 all'avvio è possibile trasferire automaticamente i file alla CPU senza alcun file dei parametri memorizzato nella schedina di memoria modificando il nome del file di programma in REPLACE.OBJ. Insieme a REPLACE.OBJ è inoltre possibile trasferire file di dati specificando i seguenti nomi di file: REPLACE.IOM, REPLCDM.IOM e REPLCE□.IOM.
6. Quando si trasferiscono progetti a una CPU della serie CS/CJ versione 3.0 o successiva da CX-Programmer versione 5.0 o successiva, è possibile memorizzare i file delle tabelle dei simboli, i file di commento e i file di indice dei programmi nella memoria dei commenti all'interno della memoria flash interna della CPU (solo se non è disponibile una memory card, la memoria file nell'area EM o altra area di memoria).
   I file di backup delle tabelle dei simboli, dei file di commento e dei file di indice dei programmi memorizzati nella memory card, nella memoria file nell'area EM o nella memoria dei commenti verranno automaticamente creati e memorizzati nella memory card.

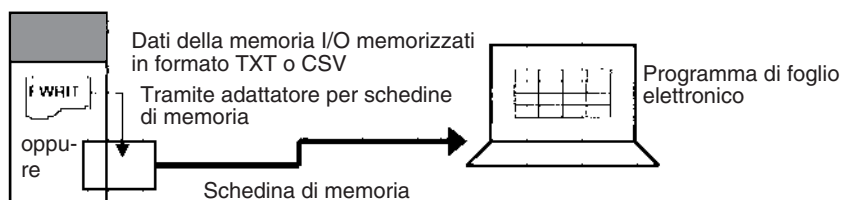
5-1-5 Ambiti di applicazione

È possibile utilizzare la memoria file per i seguenti ambiti di applicazione.

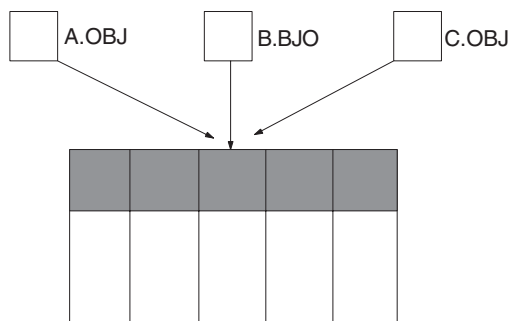
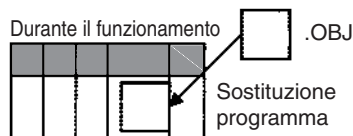
File di dati

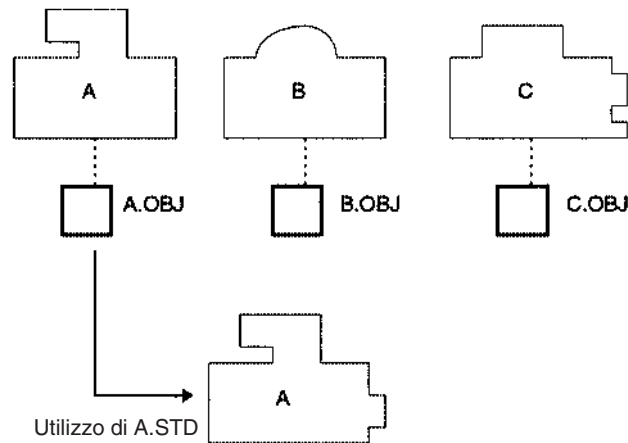
Questo ambito di applicazione prevede la memorizzazione delle impostazioni dell'area DM per Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS) nella schedina di memoria. Se il file di dati è denominato AUTOEXEC.IOM, le impostazioni memorizzate nel file verranno automaticamente trasferite all'accensione del sistema.

I dati relativi alle operazioni (tendenze, controllo qualità e altri dati) generati durante l'esecuzione del programma vengono memorizzati nella memoria file nell'area EM utilizzando l'istruzione WRITE DATA FILE (FWRIT(701)).

Nota Si consiglia di memorizzare i dati a cui si accede con maggiore frequenza, ad esempio i dati di tendenza, nella memoria file nell'area EM anziché sulla schedina di memoria.

File di dati ASCII (.TXT e .CSV)

È possibile trasferire i dati di produzione salvati sulla schedina di memoria in formato TXT o CSV a un personal computer utilizzando un adattatore per schedine di memoria e modificarli tramite un foglio elettronico (funzione non supportata dalle CPU CS1 della serie CS precedenti a EV1).

Viceversa, è possibile creare dati, quali configurazioni del Modulo di I/O speciale, in formato TXT o CSV utilizzando un foglio elettronico, memorizzarli su una schedina di memoria, quindi trasferirli nella CPU mediante l'istruzione FREAD(700) (funzione non supportata dalle CPU CS1 della serie CS precedenti a EV1).

File di programma (.OBJ)

In questo ambito di applicazione, i programmi che controllano differenti processi vengono memorizzati su singole schedine di memoria. È possibile modificare l'intera configurazione del PLC (programma, impostazioni del PLC e così via) inserendo una diversa schedina di memoria e utilizzando la funzione di trasferimento automatico all'avvio.

È possibile sostituire l'intero programma durante il funzionamento controllato dal programma stesso, senza ricorrere a un dispositivo di programmazione, utilizzando un bit di controllo dell'area ausiliaria (funzione non supportata dalle CPU CS1 della serie CS precedenti a EV1).

File dell'area dei parametri (.STD)

In questo ambito di applicazione, la configurazione del PLC, le tabelle di routing, la tabella di I/O e altri dati per determinati dispositivi o macchine vengono memorizzati su schedine di memoria. Per trasferire i dati ad altri dispositivi o macchine, è sufficiente sostituire la schedina di memoria.

File di backup

È possibile utilizzare la funzione di backup per memorizzare tutti i dati della CPU, ovvero l'intera memoria I/O, il programma e l'area dei parametri, sulla schedina di memoria senza ricorrere a un dispositivo di programmazione. Nel caso in cui si riscontrino problemi relativi ai dati della CPU, sarà possibile ripristinare immediatamente i dati di backup (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1).

File delle tabelle dei simboli

È possibile utilizzare CX-Programmer per salvare le tabelle dei simboli di programma e i commenti degli I/O in un file denominato SYMBOLS.SYM sulle schedine di memoria o nella memoria file nell'area EM.

Se si utilizza CX-Programmer versione 5.0 o successiva con una CPU della serie CS/CJ versione 3.0 o successiva, è possibile salvare i file delle tabelle dei simboli nella memoria dei commenti all'interno della memoria flash interna della CPU se non è disponibile una memory card, la memoria file nell'area EM o altra area di memoria.

File di commento

È possibile utilizzare CX-Programmer per memorizzare i commenti di rung del programma in un file di commento denominato COMMENTS.CMT sulle schedine di memoria o nella memoria file nell'area EM.

Se si utilizza una CPU della serie CS/CJ versione 3.0 o successiva, è possibile salvare i file dei commenti nella memoria dei commenti all'interno della memoria flash interna della CPU indipendentemente dalla disponibilità di una memory card o della memoria file nell'area EM.

5-2 Gestione dei file

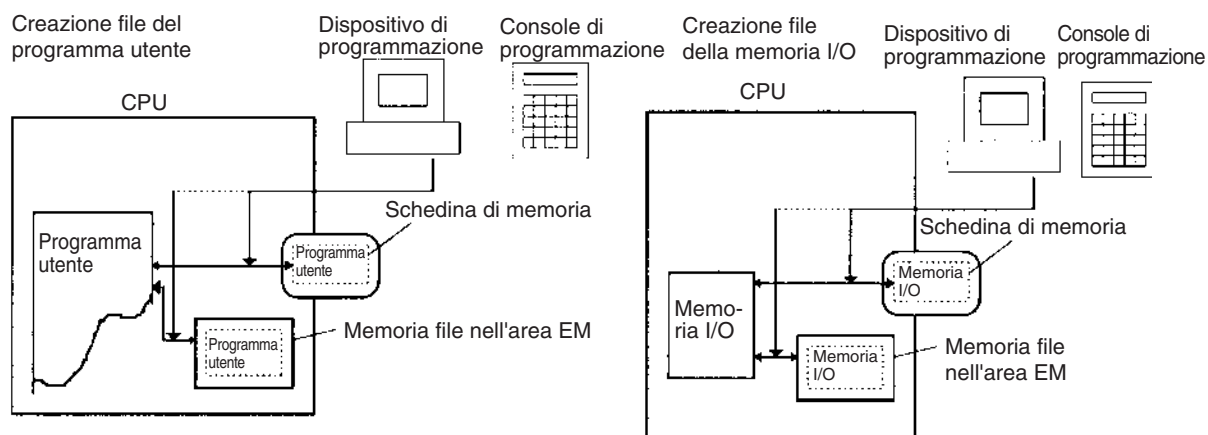
Le procedure riportate di seguito vengono utilizzate per leggere, scrivere ed eseguire altre operazioni sui file utilizzando i metodi seguenti:

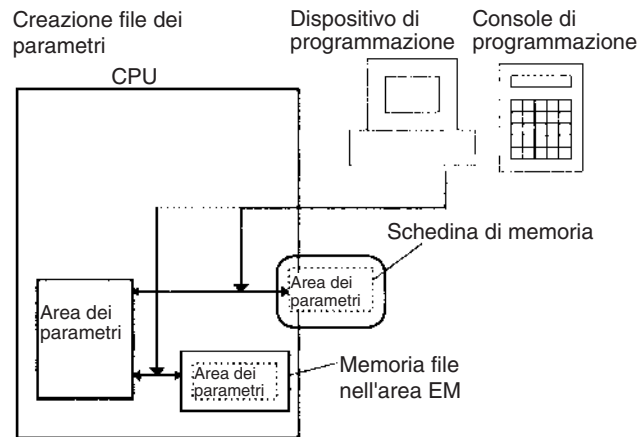
- dispositivi di programmazione
- comandi FINS
- Istruzioni FREAD(700), FWRIT(701) e CMND(490) nel programma utente (l'istruzione CMND(490) non è supportata dalle CPU CS1 serie CS precedenti a EV1)
- Sostituzione dell'intero programma mediante bit di controllo dell'area ausiliaria (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1)
- Trasferimento automatico all'avvio
- Funzione di backup (non supportata dalle CPU CS1 serie CS precedenti a EV1)

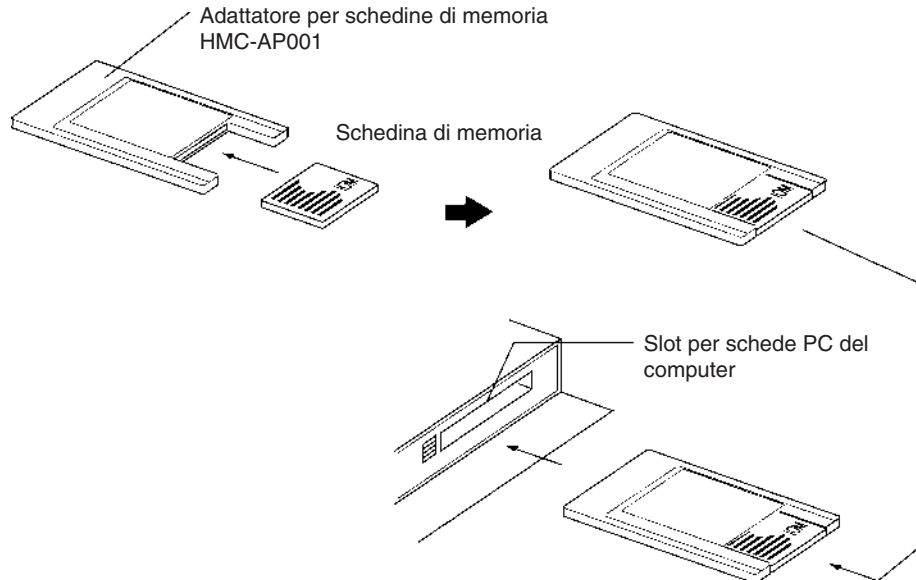
5-2-1 Dispositivi di programmazione (Console di programmazione incluse)

I dispositivi di programmazione consentono di eseguire le operazioni riportate di seguito.

Funzionamento		CX-Programmer	Console di programmazione
Lettura di file (trasferimento dalla memoria file alla CPU)		**OK**	**OK**
Scrittura di file (trasferimento dalla CPU alla memoria file)		**OK** (vedere nota)	**OK** (vedere nota)
Confronto tra file (confronto dei file presenti nella CPU e nella memoria file)		Non previsto	**OK**
Formattazione della memoria file	schedine di memoria	**OK**	**OK**
	File EM	**OK**	**OK**
Modifica dei nomi dei file		**OK**	Non previsto
Lettura dei dati nella memoria file		**OK**	Non previsto
Eliminazione di file		**OK**	**OK**
Copia di file		**OK**	Non previsto
Creazione/eliminazione di sottodirectory		**OK**	Non previsto

Nota Con le CPU della serie CS/CJ versione 2.0 o successiva è possibile utilizzare la protezione da lettura tramite password per impedire la scrittura di un file di programma nella memoria file, ad esempio una schedina di memoria o la memoria file nell'area EM.

Nota

1. Creare le etichette di volume necessarie utilizzando Esplora risorse di Windows.
2. La memoria file supporta la formattazione rapida di Windows. Se si verificano errori di formattazione nelle schedine di memoria, è possibile formattarle utilizzando l'opzione di formattazione completa.
3. L'ora e la data dei file scritti per i trasferimenti dalla CPU alla memoria file vengono ricavate dall'orologio della CPU.

Nota Con le CPU della serie CS/CJ versione 2.0 o successiva e CX-Programmer versione 4.0 o successiva è possibile impedire la creazione opzionale di un file di programma di backup (.OBJ) durante la registrazione di una password per l'intero programma utente o per specifici task. Per ulteriori informazioni, fare riferimento a *Protezione da scrittura del programma* nella sezione *1-4-2 Migliore protezione da lettura mediante password* del *Manuale dell'operatore della serie CJ*.

È possibile installare una schedina di memoria nello slot per schede PLC di un computer utilizzando l'adattatore HMC-AP001 (acquistabile separatamente). L'installazione di una schedina di memoria nel computer consente la lettura dei file contenuti nella schedina e la scrittura di file per mezzo di altri programmi, ad esempio Esplora risorse.

CX-Programmer

Per eseguire operazioni relative alla memoria file, attenersi alla procedura descritta.

1,2,3...

1. Fare doppio clic sull'icona della schedina di memoria nella finestra del progetto con la CPU in linea. Viene visualizzata la finestra Memory Card (Schedina di memoria).
2. Per trasferire i dati dalla CPU alla memoria file, selezionare l'area del programma, l'area della memoria I/O o l'area dei parametri nell'area di lavoro del progetto, selezionare ***Transfer*** (Trasferisci) dalla memoria file,

quindi selezionare l'opzione di trasferimento alla schedina di memoria o alla memoria file nell'area EM.

oppure Per trasferire i dati dalla memoria file alla CPU, selezionare il file nella memoria file e trascinarlo sull'area del programma, l'area della memoria I/O o l'area dei parametri nell'area di lavoro del progetto.

Nota Per creare e leggere i file delle tabelle dei simboli (SYMBOLS.SYM) e i file di commento (COMMENTS.CMT) in CX-Programmer, utilizzare le operazioni di trasferimento del progetto.

Console di programmazione

È possibile eseguire le operazioni riportate di seguito.

Voce 1	Voce 2	Voce 3	Voce 4	Voce 5
0: invio	0: da PLC a schedina di memoria	Selezionare OBJ, CIO, HR, WR, AR, DM, EM o STD.	Impostare gli indirizzi iniziale e finale per il trasferimento.	Tipo di supporto, nome del file
	1: da schedina di memoria a PLC	Selezionare OBJ, CIO, HR, WR, AR, DM, EM o STD.	Impostare gli indirizzi iniziale e finale per il trasferimento.	Tipo di supporto, nome del file
1: verifica		Selezionare OBJ, CIO, HR, WR, AR, DM, EM o STD.	Impostare gli indirizzi iniziale e finale per il confronto.	Tipo di supporto, nome del file
2: inizializzazione		Immettere 9713 (schedina di memoria) o 8426 (memoria file nell'area EM).	---	---
3: eliminazione		Selezionare OBJ, CIO, HR, WR, AR, DM, EM o STD.	Tipo di supporto, nome del file	---

Nota Nella seguente tabella sono riportati i tipi di file.

Simbolo	Tipo di file	
OBJ	File di programma (.OBJ)	
CIO	File di dati (.IOM)	Area CIO
HR		Area di ritenzione
WR		Area WR
AR		Area ausiliaria
DM		Area DM
EM0_		Area EM
STD	File dei parametri (.STD)	

Precauzioni per il confronto di dati dopo il trasferimento di file dei parametri

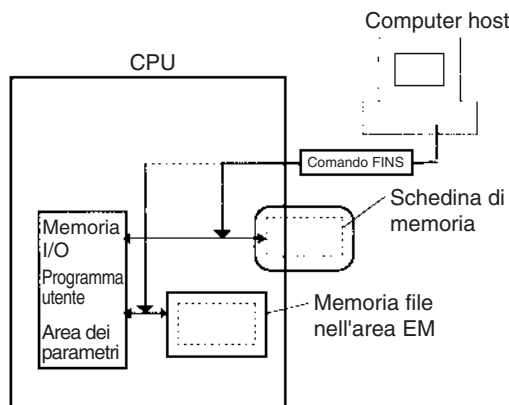
È possibile che sulla Console di programmazione si verifichino degli errori quando si confrontano dati dei parametri tra file prima e dopo il trasferimento se i file dei parametri (STD) creati in una CPU della serie CJ vengono salvati nella memory card all'interno di un'altra CPU della serie CJ con versione diversa. Gli errori che si possono verificare per le combinazioni di diverse versioni della CPU sono riportati nella seguente tabella.

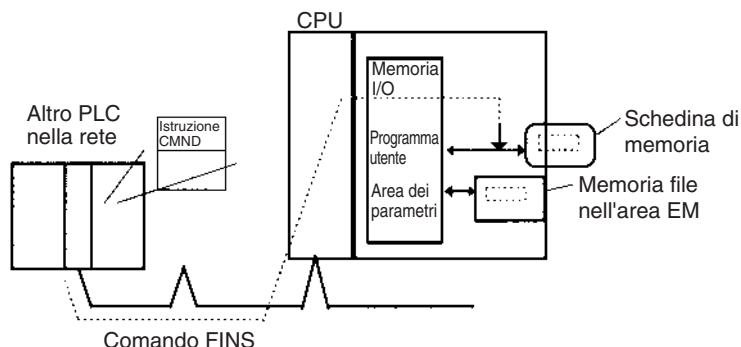
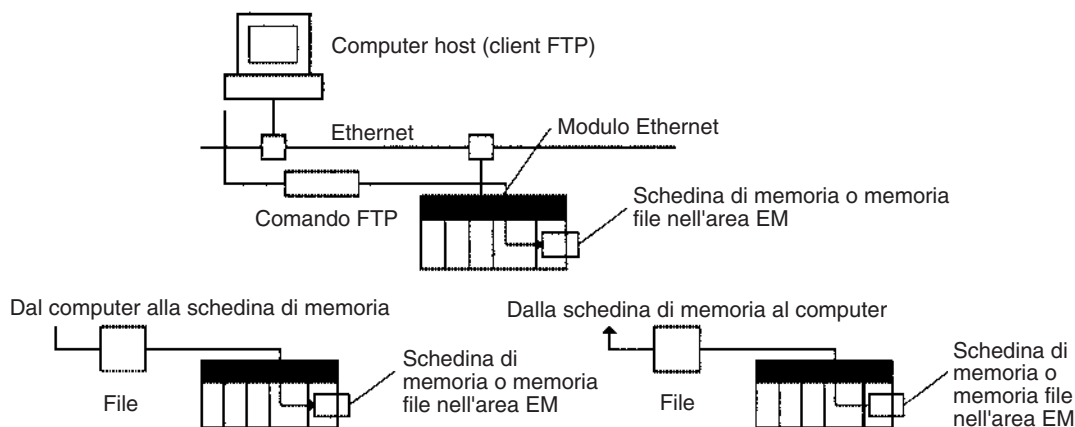
CPU di origine		Stato di allocazione degli I/O di origine	CPU di destinazione		
			Versione della CPU a cui verranno trasferiti i file dei parametri		
			CPU precedenti alla versione 2.0	CPU versione 2.0	CPU versione 3.0 o successiva
Versione della CPU in cui sono stati creati i file dei parametri	**CPU precedenti alla versione 2.0**	Allocazione automatica	Verifica possibile	Verifica possibile	Errore di verifica
		Definita dall'utente			Errore di verifica
	CPU versione 2.0	Allocazione automatica			Errore di verifica
		Definita dall'utente			Verifica possibile
	CPU versione 3.0 o successiva	Allocazione automatica		Errore di verifica	Verifica possibile
		Definita dall'utente		Verifica possibile	
Versione della CPU utilizzata per creare i file per il trasferimento automatico all'avvio	**CPU precedenti alla versione 2.0**	Allocazione automatica	Passa a quella definita dall'utente	(Uguale) Allocazione automatica	Passa a quella definita dall'utente
		Definita dall'utente	(Uguale) Definita dall'utente	Passa all'allocazione automatica (vedere nota 1)	(Uguale) Definita dall'utente
	CPU versione 2.0	Allocazione automatica	Passa a quella definita dall'utente	(Uguale) Allocazione automatica	Passa al funzionamento utente
		Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente
	CPU versione 3.0 o successiva	Allocazione automatica	Passa a quella definita dall'utente	(Uguale) Allocazione automatica	(Uguale) Allocazione automatica
		Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente

5-2-2 Comandi FINS

Inviando alla CPU il comando FINS appropriato, è possibile eseguire le operazioni relative alla memoria file indicate di seguito. Tali operazioni sono simili a quelle eseguibili mediante le funzioni del dispositivo di programmazione.

Invio di comandi FINS tramite Host Link

Con un computer collegato tramite un sistema Host Link è possibile inviare un comando FINS con un'intestazione e un terminatore Host Link.

Invio di comandi FINS da un altro PLC della rete

È possibile inviare un comando FINS da un altro PLC collegato in rete utilizzando l'istruzione CMND(490).

Nota È possibile utilizzare un computer su una rete Ethernet per leggere e scrivere file (schedine di memoria o memoria file nell'area EM) su una CPU tramite un Modulo Ethernet. È inoltre possibile scambiare dati se il computer host opera come client FTP e il PLC della serie CS/CJ come server FTP.

I comandi FINS riportati di seguito possono essere utilizzati per eseguire svariate operazioni, compresa la lettura e scrittura di file.

Comando	Nome	Descrizione
2201 (esa-decimale)	FILE NAME READ	Legge i dati nella memoria file.
2202 (esa-decimale)	SINGLE FILE READ	Legge una determinata quantità di dati del file a partire da una posizione specificata all'interno di un singolo file.
2203 (esa-decimale)	SINGLE FILE WRITE	Scrive una determinata quantità di dati del file a partire da una posizione specificata all'interno di un singolo file.
2204 (esa-decimale)	FILE MEMORY FORMAT	Formatta, ovvero inizializza, la memoria file.
2205 (esa-decimale)	FILE DELETE	Elimina i file specificati memorizzati nella memoria file.
2207 (esa-decimale)	FILE COPY	Copia i file da una memoria file a un'altra memoria file.
2208 (esa-decimale)	FILE NAME CHANGE	Modifica il nome di un file.
220A (esa-decimale)	MEMORY AREA FILE TRANSFER	Trasferisce o confronta i dati tra l'area della memoria I/O e la memoria file.
220B (esa-decimale)	PARAMETER AREA FILE TRANSFER	Trasferisce o confronta i dati tra l'area dei parametri e la memoria file.
220C (esa-decimale)	PROGRAM AREA FILE TRANSFER	Trasferisce o confronta i dati tra l'area UM (memoria utente) e la memoria file.
2215 (esa-decimale)	CREATE/DELETE SUBDIRECTORY	Crea ed elimina sottodirectory.

Nota Per specificare la data di creazione dei file nella memoria file con i comandi 220A, 220B, 220C e 2203, viene utilizzata l'ora dell'orologio interno della CPU.

5-2-3 Istruzioni FREAD(700), FWRIT(701) e CMND(490)

È possibile utilizzare l'istruzione FWRIT(701) (WRITE DATA FILE) per creare un file di dati contenente i dati della memoria I/O specificata in una schedina di memoria o nella memoria file nell'area EM. È inoltre possibile sovrascrivere o aggiungere dati in qualsiasi punto all'interno di file già esistenti.

L'istruzione FREAD(700) (READ DATA FILE) consente di leggere i dati della memoria I/O da una posizione specifica in un file di dati memorizzato su una schedina di memoria o nella memoria file nell'area EM e scriverli nella porzione specificata della memoria I/O. È possibile leggere i dati a partire da qualsiasi punto all'interno del file specificato.

Nota Queste istruzioni non comportano il trasferimento dell'intero file, bensì solo il trasferimento della quantità di dati specificata a partire dalla posizione iniziale indicata all'interno del file.

L'istruzione CMND(490) (DELIVER COMMAND) viene utilizzata per inviare un comando FINS alla CPU allo scopo di eseguire operazioni sui file. È possibile eseguire operazioni quali formattazione, eliminazione, copia e ridenominazione dei file presenti nella schedina di memoria o nella memoria file nell'area EM (funzioni non supportate dalle CPU CS1 della serie CS precedenti a EV1).

FREAD(700)/FWRIT(701): esegue trasferimenti tra la memoria I/O e la memoria file

CPU

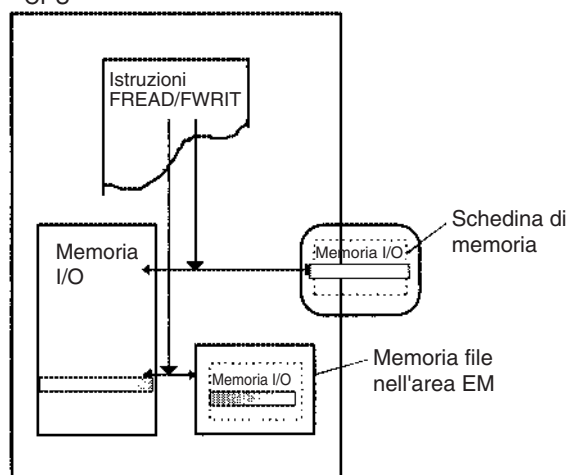
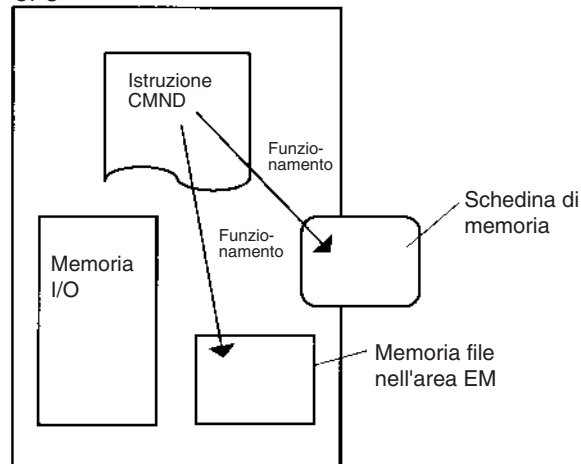
Istruzioni FREAD/FWRIT

Memoria I/O

Memoria I/O

Schedina di memoria

Memoria I/O

Memoria file nell'area EM

CMND(490): operazioni sulla memoria file (non supportate dalle CPU serie CS precedenti a EV1)

Istruzioni FREAD(700) e FWRIT(701)

FREAD(700) e FWRIT(701) consentono di trasferire i dati tra la memoria I/O e la memoria file. Tutte le CPU CJ consentono di trasferire dati binari (file .IOM), mentre le CPU V1 consentono di trasferire anche file ASCII (file .TXT e .CSV).

Nome	Codice mnemonico	Descrizione
READ DATA FILE	FREAD(700)	Legge i dati del file o gli elementi dei dati specificati nella memoria I/O specificata.
WRITE DATA FILE	FWRIT(701)	Utilizza i dati dell'area specificata della memoria I/O per creare il file di dati specificato.

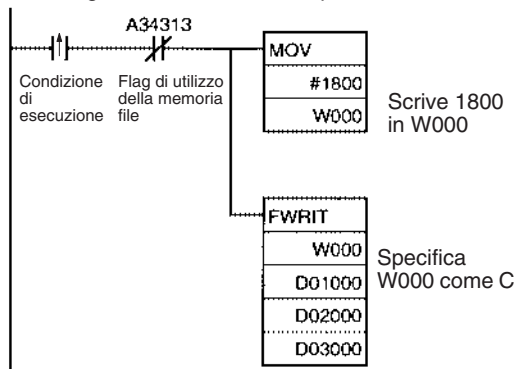
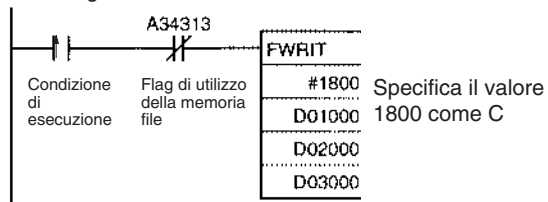
Trasferimento di file ASCII (non supportato dalle CPU CS1 serie CS precedenti a EV1)

Oltre ai file binari, è possibile trasferire anche file ASCII in modo che la terza e quarta cifra dell'operando del canale di controllo dell'istruzione (C) indichino il tipo di file di dati in corso di trasferimento e il numero di campi tra i ritorni a capo.

Bit in C	Impostazioni	Limitazioni del dispositivo di programmazione
Da 12 a 15	Tipo di dati 0: binario (.IOM) 1: canali non delimitati (.TXT) 2: canali doppi non delimitati (.TXT) 3: canali delimitati da virgole (.CSV) 4: canali doppi delimitati da virgole (.CSV) 5: canali delimitati da tabulazioni (.TXT) 6: canali doppi delimitati da tabulazioni (.TXT)	Se si utilizza CX-Programmer 1.1 o una versione precedente, è possibile specificare direttamente solo il valore esadecimale 0 (file .IOM). Se si utilizza CX-Programmer 1.2 o versione successiva, oppure una Console di programmazione, è possibile impostare i bit del canale di controllo su valori esadecimali compresi tra 0 e 6.
Da 08 a 11	Ritorni a capo 0: nessun ritorno a capo 8: ritorno a capo ogni 10 campi 9: ritorno a capo a ogni campo A: ritorno a capo ogni 2 campi B: ritorno a capo ogni 4 campi C: ritorno a capo ogni 5 campi D: ritorno a capo ogni 16 campi	Se si utilizza CX-Programmer 1.1 o una versione precedente, oppure una Console di programmazione, è possibile specificare direttamente solo il valore esadecimale 0 (nessun ritorno a capo). Se si utilizza CX-Programmer 1.2 o versione successiva, è possibile impostare i bit del canale di controllo su 0 esadecimale oppure sui valori esadecimali compresi tra 8 e D.

CX-Programmer 1.1 o versione precedente: impostazione indiretta del canale di controllo

Se si utilizza CX-Programmer 1.1 o una versione precedente, non è possibile trasferire file ASCII con le istruzioni FREAD(700) e FWRIT(701) quando si immette una costante per il canale di controllo per specificare il tipo di dati e il trattamento dei ritorni a capo. Se si utilizza una costante, sarà possibile trasferire solo dati binari senza ritorni a capo.

È tuttavia possibile trasferire file ASCII mediante le istruzioni FREAD(700) e FWRIT(701) impostando indirettamente il canale di controllo. A tale scopo, definire l'impostazione desiderata del canale di controllo in un canale e specificare quest'ultimo come canale di controllo in FREAD(700) o FWRIT(701), come indicato a sinistra nel seguente diagramma.

Nota Per specificare la data di creazione dei file nella memoria file con l'istruzione FWRIT(701), viene utilizzata l'ora dell'orologio interno della CPU.

Poiché è possibile eseguire una sola operazione per volta sulla memoria file, FREAD(700) e FWRIT(701) non devono essere eseguite quando è in corso una qualunque delle seguenti operazioni:

1,2,3...
1. Esecuzione di FREAD(700) o FWRIT(701)
2. Esecuzione di CMND(490) per inviare un comando FINS alla CPU
3. Sostituzione dell'intero programma tramite operazioni dei bit di controllo dell'area ausiliaria
4. Esecuzione di un'operazione di backup semplice

Utilizzare il flag di utilizzo della memoria file (A34313) per controllare le istruzioni relative alla memoria file in modo da impedirne l'esecuzione quando è in corso un'altra operazione.

Quando viene eseguita l'istruzione FREAD(700), il flag di errore di lettura file (A34310) viene attivato e l'esecuzione viene interrotta se il file specificato contiene il tipo di dati errato o se i dati del file sono danneggiati. Per i file TXT o CSV, il codice carattere deve essere in formato esadecimale e i delimitatori devono essere inseriti ogni 4 cifre per i dati a canale singolo e ogni 8 cifre per i dati a canale doppio. I dati verranno letti finché non verrà rilevato un carattere non valido.

Bit e canali ausiliari correlati

Nome	Indirizzo	Funzionamento
Tipo di schedina di memoria	Da A34300 ad A34302	Indica il tipo di schedina di memoria eventualmente installata.
Flag di errore di formattazione della memoria file nell'area EM	A34306	Viene attivato se si verifica un errore di formattazione nel primo banco dell'area EM assegnato alla memoria file. Viene disattivato quando la formattazione viene completata normalmente.
Flag di errore di formattazione della schedina di memoria	A34307	Viene attivato quando la schedina di memoria non è formattata o si verifica un errore di formattazione.
Flag di errore di scrittura file	A34308	Viene attivato quando si verifica un errore durante la scrittura nel file.
Flag di scrittura file non possibile	A34309	Viene attivato quando non è possibile scrivere dati perché il file è protetto da scrittura o la quantità di memoria disponibile è insufficiente.
Flag di errore di lettura file	A34310	Viene attivato quando non è possibile leggere un file perché i dati sono danneggiati o il file contiene dati di tipo errato.
Flag di file non trovato	A34311	Viene attivato quando non è possibile leggere i dati perché il file specificato non esiste.
Flag di utilizzo della memoria file	A34313	Viene attivato in ciascuno dei seguenti casi: La CPU sta elaborando un comando FINS inviato tramite istruzione CMND(490). È in esecuzione un'istruzione FREAD(700) o FWRIT(701). È in corso la sovrascrittura del programma tramite l'utilizzo di un bit di controllo dell'area ausiliaria. È in corso un'operazione di backup semplice.
Flag di accesso al file	A34314	Viene attivato quando è in corso l'accesso ai dati del file.
Flag di rilevamento della schedina di memoria	A34315	Viene attivato quando viene rilevata una schedina di memoria. (non supportata dalle CPU CS1 serie CS precedenti a EV1)
Numero di elementi da trasferire	Da A346 ad A347	Questi canali indicano il numero di canali o campi che rimangono da trasferire (32 bit). Quando un file binario (.IOM) è in corso di trasferimento, tale numero diminuisce man mano che i canali vengono letti. Quando un file TXT o CSV è in corso di trasferimento, tale numero diminuisce man mano che i campi vengono trasferiti.

CMND(490): DELIVER COMMAND

È possibile utilizzare l'istruzione CMND(490) per inviare un comando FINS alla CPU locale allo scopo di eseguire operazioni sulla memoria file, ad esempio un'operazione di formattazione o di eliminazione di file. Quando si invia un comando FINS relativo alla memoria file al PLC locale, è necessario impostare i seguenti valori per i canali di controllo di CMND(490):

1,2,3...

1. Impostare l'indirizzo di rete di destinazione su 00 (rete locale) in C+2.
2. Impostare l'indirizzo del modulo di destinazione su 00 (CPU del PLC) e il nodo di destinazione su 00 (all'interno del nodo locale) in C+3.
3. Impostare il numero di tentativi su 0 in C+4. Poiché l'impostazione del numero di tentativi non è un'opzione valida, impostarla su 0.

Comandi FINS correlati alla memoria file

Per informazioni sui comandi FINS, fare riferimento alla sezione *5-2-2 Comandi FINS*.

Nota Esistono altri comandi FINS correlati alla memoria file che è possibile eseguire e che non sono riportati nella tabella fornita di seguito. Per ulteriori informazioni sui comandi FINS, fare riferimento al manuale *Communications Command Reference Manual* (W342).

L'istruzione CMND(490) non può essere eseguita sulla CPU locale se è già in esecuzione un'altra istruzione CMND(490) su un'altra CPU, se è in esecuzione un'istruzione FREAD(700) o FWRIT(701), se è in corso la sostituzione del programma mediante bit di controllo dell'area ausiliaria oppure se è in corso un'operazione di backup semplice. Accertarsi di utilizzare il flag di utilizzo della memoria file, ossia A34313, come condizione NC (normalmente chiusa) per impedire l'esecuzione di CMND(490) quando è in corso un'altra operazione sulla memoria file.

Se non è possibile eseguire CMND(490) per la CPU locale, il flag di errore verrà attivato.

Bit e canali ausiliari correlati

Nome	Indirizzo	Funzionamento
Flag di utilizzo della memoria file	A34313	Viene attivato in ciascuno dei seguenti casi: • La CPU sta elaborando un comando FINS inviatosi tramite l'istruzione CMND(490). • È in esecuzione un'istruzione FREAD(700) o FWRIT(701). • È in corso la sovrascrittura del programma tramite l'utilizzo di un bit di controllo dell'area ausiliaria. • È in corso un'operazione di backup semplice.
Flag di rilevamento della schedina di memoria	A34315	Viene attivato quando viene rilevata una schedina di memoria. (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1).

L'esempio che segue illustra come utilizzare CMND(490) per creare un sottodirectory nella schedina di memoria.

Quando 000000 e A20207 sono impostati su ON e A34313 su OFF, l'istruzione CMND(490) invia il comando FINS 2215 (CREATE/DELETE SUBDIRECTORY) alla CPU locale e la risposta viene memorizzata in D00100 e D00101.

In questo esempio il comando FINS crea la sottodirectory CS1 nella directory OMRON sulla schedina di memoria della CPU. La risposta è costituita da 2 byte contenenti il codice di comando (2215) e da 2 byte contenenti il codice di risposta.

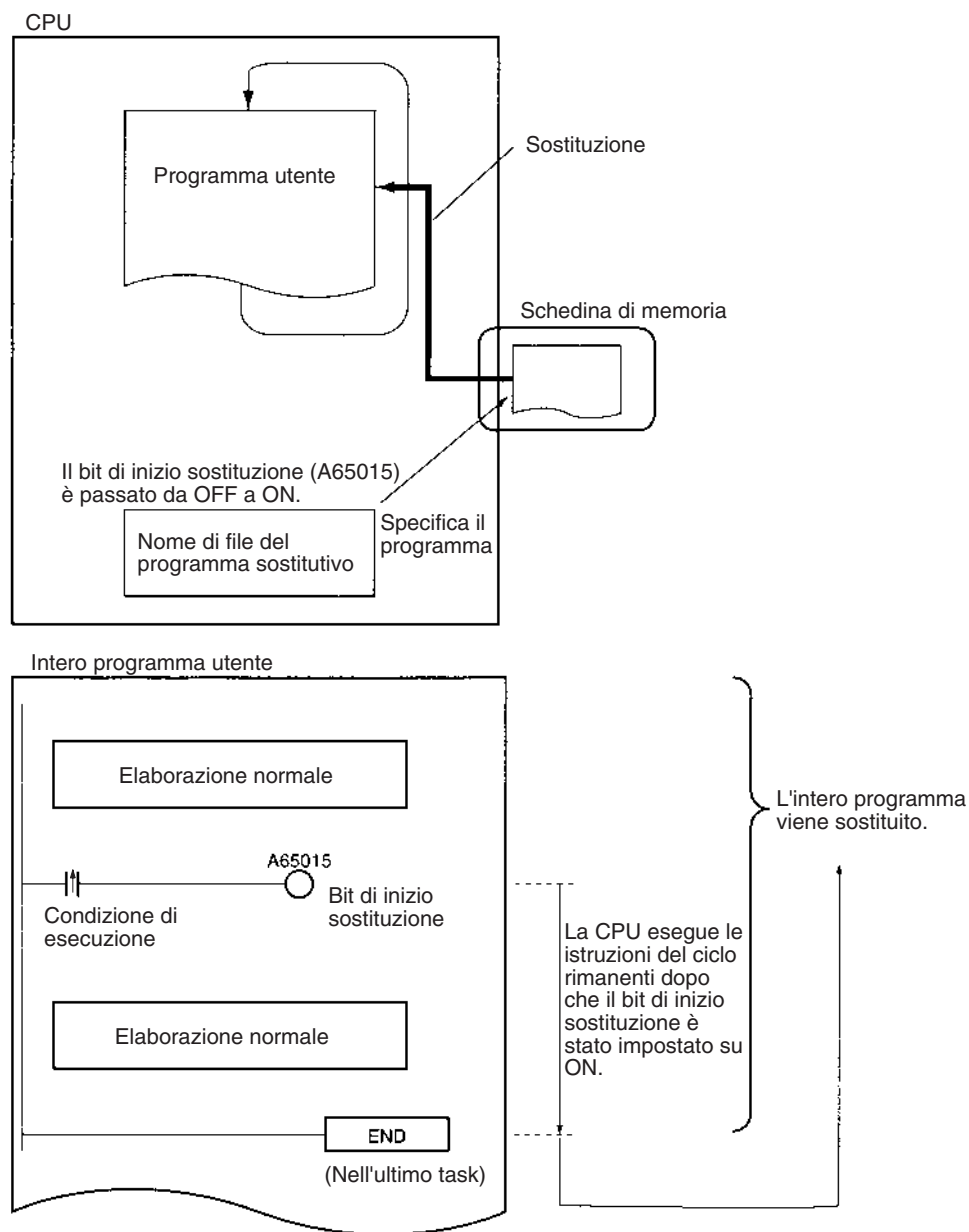
		15		8 7	0	
S:	D00006	2	2	1	5	Codice comando: 2215 esadecimale (CREATE/DELETE SUBDIRECTORY)
S+1:	D00007	8	0	0	0	Numero disco: 8000 esadecimale (schedina di memoria)
S+2:	D00008	0	0	0	0	Parametro: 0000 esadecimale (creazione di una sottodirectory)
S+3:	D00009	4	3	5	3	Nome sottodirectory: CS1□□□□□.□□□ (□: uno spazio)
S+4:	D00010	3	1	2	0	
S+5:	D00011	2	0	2	0	
S+6:	D00012	2	0	2	0	
S+7:	D00013	2	E	2	0	
S+8:	D00014	2	0	2	0	
S+9:	D00015	0	0	0	6	Lunghezza directory: 0006 esadecimale (6 caratteri)
S+10:	D00016	5	C	4	F	Percorso di directory: \OMRON
S+11:	D00017	4	D	5	2	
S+12:	D00018	4	F	4	E	

		15		8 7	0	
C:	D00000	0	0	1	A	Numero di byte dei dati di comando: 001A esadecimale (26 byte)
C+1:	D00001	0	0	0	4	Numero di byte dei dati di risposta: 0004 esadecimale (4 byte)
C+2:	D00002	0	0	0	0	Indirizzo di destinazione: 0000 esadecimale (rete locale)
C+3:	D00003	0	0	0	0	00 esadecimale (nodo locale) e 00 esadecimale (CPU)
C+4:	D00004	0	7	0	0	Risposta richiesta, porta di comunicazione 7, 0 tentativi
C+5:	D00005	0	0	0	0	Tempo di monitoraggio della risposta: FFFF esadecimale (6553,5 s)

Nota Oltre ai comandi FINS correlati alle operazioni sulla memoria file riportati nella tabella, ne esistono altri che è possibile inviare al PLC locale. Utilizzare il flag di utilizzo della memoria file per impedire l'esecuzione simultanea anche di questi altri comandi FINS.

5-2-4 Sostituzione dell'intero programma durante il funzionamento

(funzione non supportata dalle CPU CS1 serie CS precedenti a EV1).

È possibile sostituire l'intero programma durante il funzionamento (modalità RUN o MONITOR) attivando il bit di inizio sostituzione (A65015). Il file specificato verrà letto dalla schedina di memoria e sostituirà il programma eseguibile alla fine del ciclo corrente. Per sostituire il programma durante il funzionamento, è necessario registrare in anticipo la password del programma (A651) e il nome del file di programma (da A654 a A657) e verificare che il file di programma specificato si trovi sulla schedina di memoria.

È anche possibile procedere alla sostituzione quando l'esecuzione del programma viene interrotta (modalità PROGRAM) attivando il bit di inizio sostituzione da un dispositivo di programmazione.

Nota Non è possibile leggere il file del programma sostitutivo dalla memoria file nell'area EM.

È possibile attivare il bit di inizio sostituzione (A65015) in qualsiasi posizione (indirizzo del programma) del programma. La CPU eseguirà le istruzioni rimanenti del ciclo dopo che lo stato del bit di inizio sostituzione sarà passato da OFF a ON.

Durante la sostituzione, il programma non viene eseguito. Dopo la sostituzione del programma, il funzionamento riprende come accade quando la CPU passa dalla modalità PROGRAM alla modalità RUN o MONITOR.

Il programma verrà sostituito al termine del ciclo in cui viene attivato il bit di inizio sostituzione, ovvero dopo che sarà stata eseguita l'istruzione END(001) nell'ultimo task del programma.

Nota

1. Per mantenere lo stato dei dati della memoria I/O durante la sostituzione del programma, attivare il bit di ritenzione IOM (A50012).

   Per mantenere lo stato dei bit impostati o ripristinati in modo forzato durante la sostituzione del programma, attivare il bit di ritenzione dello stato forzato (A50013).
2. Se il bit di ritenzione IOM (A50012) è già attivato prima della sostituzione del programma, lo stato dei bit della memoria I/O verrà mantenuto dopo la sostituzione del programma. Verificare che i carichi esterni funzionino in modo corretto con i medesimi dati della memoria I/O.

   Analogamente, se il bit di ritenzione dello stato forzato (A50013) è già attivato prima della sostituzione del programma, lo stato dei bit impostati o ripristinati in modo forzato verrà mantenuto dopo la sostituzione del programma. Verificare che i carichi esterni funzionino in modo corretto con i medesimi bit impostati o ripristinati in modo forzato.

File sostitutivo

Il file di programma di cui si è specificato il nome nei canali da A654 a A657 verrà letto dalla schedina di memoria e andrà a sostituire il programma esistente alla fine del ciclo in cui viene attivato il bit di inizio sostituzione (A65015).

File	Nome file ed estensione	Specifica del nome del file sostitutivo (********)
File di programma	********.OBJ	Scrivere il nome del file di programma sostitutivo nei canali da A654 a A657 prima della sostituzione.

Requisiti per la sostituzione del programma

Affinché sia possibile sostituire il programma durante il funzionamento, è necessario che siano state soddisfatte le seguenti condizioni:

- La password del programma (A5A5) è stata scritta in A651.
- Il file di programma specificato nei canali da A654 a A657 (nome del file di programma) si trova nella directory principale della schedina di memoria.
- La schedina di memoria è stata rilevata dalla CPU (A34315 ON).
- Non si sono verificati errori fatali.
- Non è in esecuzione alcuna operazione sulla memoria file (A34313 OFF).
- Non è in corso alcuna operazione di scrittura dati nell'area del programma.
- I diritti di accesso sono disponibili (ad esempio, non è in corso un trasferimento di dati da CX-Programmer al PLC).

Nota È possibile trasferire il programma in qualsiasi modalità operativa.

Funzionamento della CPU durante la sostituzione del programma

Durante la sostituzione del programma, la CPU presenterà il seguente comportamento:

- Esecuzione del programma: interrotta
- Monitoraggio del tempo di ciclo: nessun monitoraggio

Operazioni in corso durante e dopo la sostituzione del programma

Quando si attiva il bit di ritenzione IOM (A50012), vengono mantenuti i dati nelle seguenti aree di memoria: area CIO, area di lavoro (W), flag di completamento del temporizzatore (T), registri indice (IR), registri dati (DR) e numero del banco EM corrente.

Nota Durante la sostituzione del programma, i valori attuali del temporizzatore vengono azzerati.

Se durante il trasferimento del programma il bit di ritenzione IOM è attivato, i carichi in uscita prima della sostituzione del programma continueranno anche dopo la sostituzione. Accertarsi che dopo la sostituzione del programma i carichi esterni funzionino in modo corretto.

Se il bit di ritenzione dello stato forzato (A50013) è attivato, durante la sostituzione del programma lo stato dei bit impostati o ripristinati in modo forzato verrà mantenuto.

Gli interrupt verranno mascherati.

Se è in corso la registrazione dei dati, tale operazione verrà interrotta.

Le condizioni di istruzione (interblocchi, interruzioni ed esecuzione del programma a blocchi) verranno inizializzate.

I flag comparativi verranno inizializzati a seconda che il bit di ritenzione IOM sia attivato o disattivato.

Operazioni eseguite dopo la sostituzione del programma

Lo stato dei task ciclici dipende dalle relative proprietà di avvio. Tale stato corrisponde allo stato ottenuto quando il PLC passa dalla modalità PROGRAM alla modalità RUN o MONITOR.

Il flag del primo ciclo (A20011) avrà stato ON per la durata di un ciclo dopo la ripresa dell'esecuzione del programma. Tale stato corrisponde allo stato ottenuto quando il PLC passa dalla modalità PROGRAM alla modalità RUN o MONITOR.

Tempo necessario per la sostituzione del programma

Dimensioni dell'intero programma	Tempo del ciclo operativo della periferica impostato nella configurazione del PLC	Tempo (approssimato) necessario per la sostituzione del programma
60.000 step	Predefinito (4% del tempo di ciclo)	6 s
250.000 step		25 s

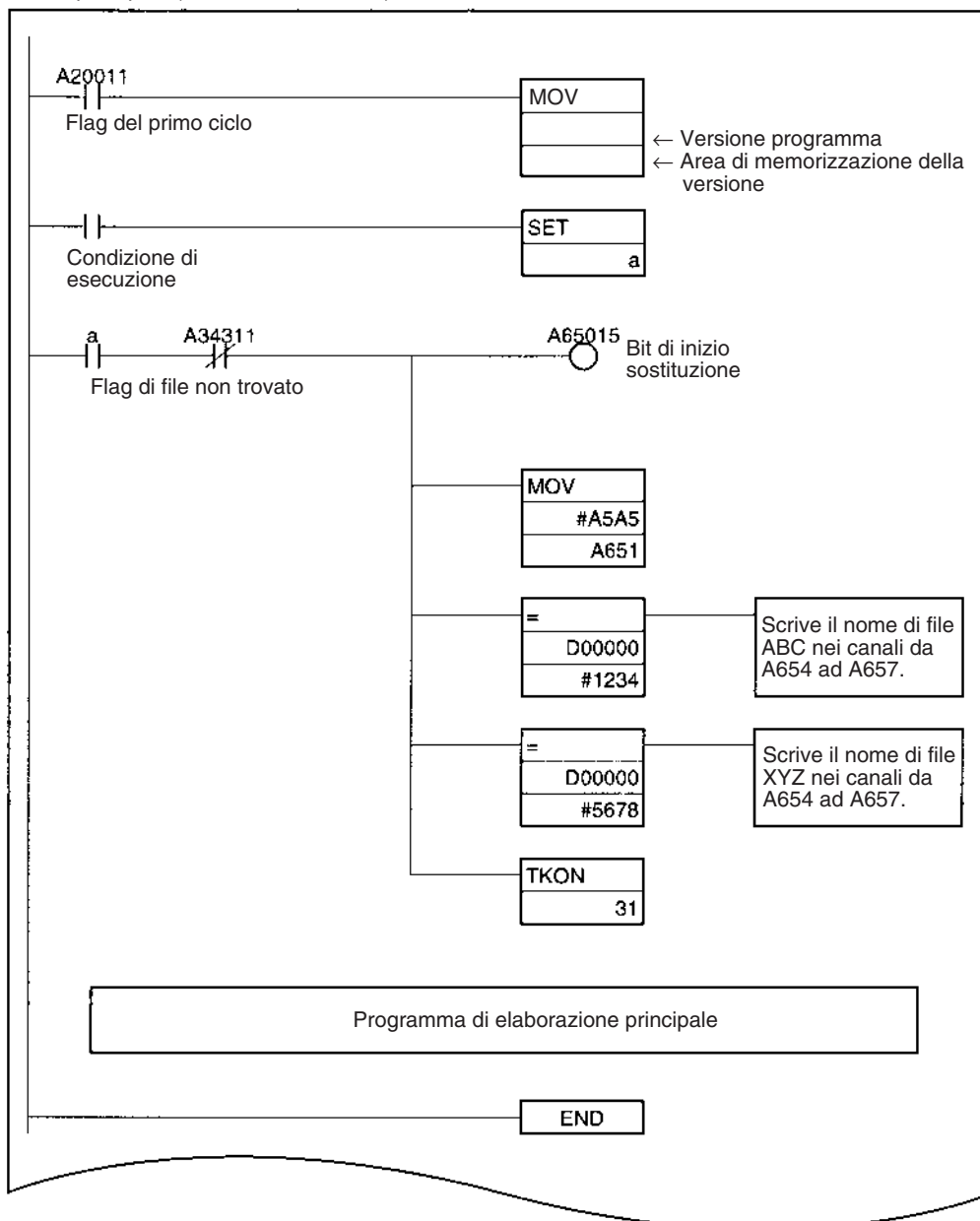
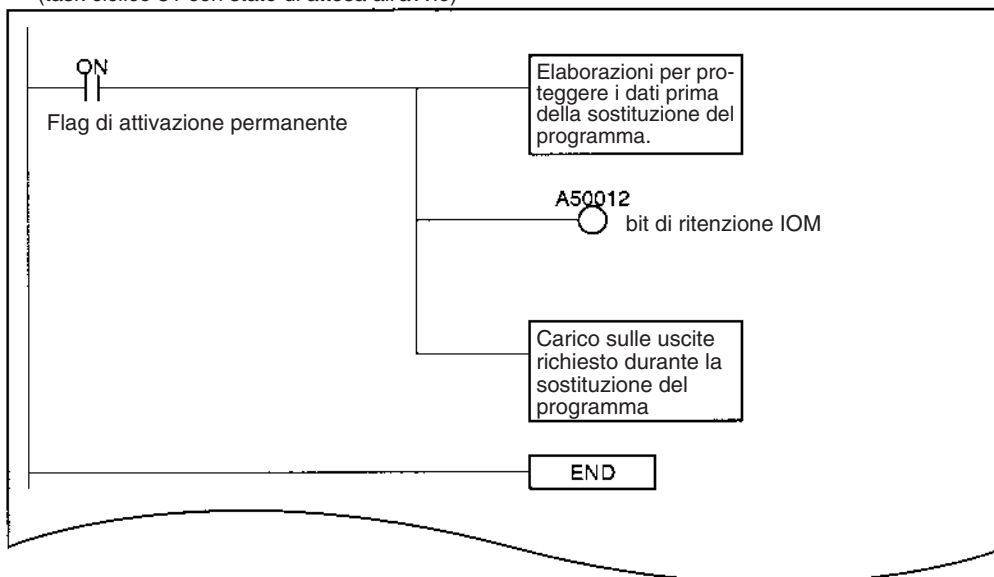
Bit e canali ausiliari correlati

Nome	Indirizzo	Funzionamento
Flag di utilizzo della memoria file	A34313	Viene attivato in ciascuno dei seguenti casi: La CPU ha inviato un comando FINS utilizzando CMND(490). È in esecuzione un'istruzione FREAD(700) o FWRIT(701). È in corso la sovrascrittura del programma tramite l'utilizzo di un bit di controllo dell'area ausiliaria (A65015). È in corso un'operazione di backup semplice.
Flag di rilevamento della schedina di memoria (non supportato dalle CPU CS1 serie CS precedenti a EV1)	A34315	Viene attivato quando viene rilevata una schedina di memoria.
bit di ritenzione IOM	A50012	Quando il bit è attivato, il contenuto della memoria I/O viene mantenuto durante la sostituzione del programma.

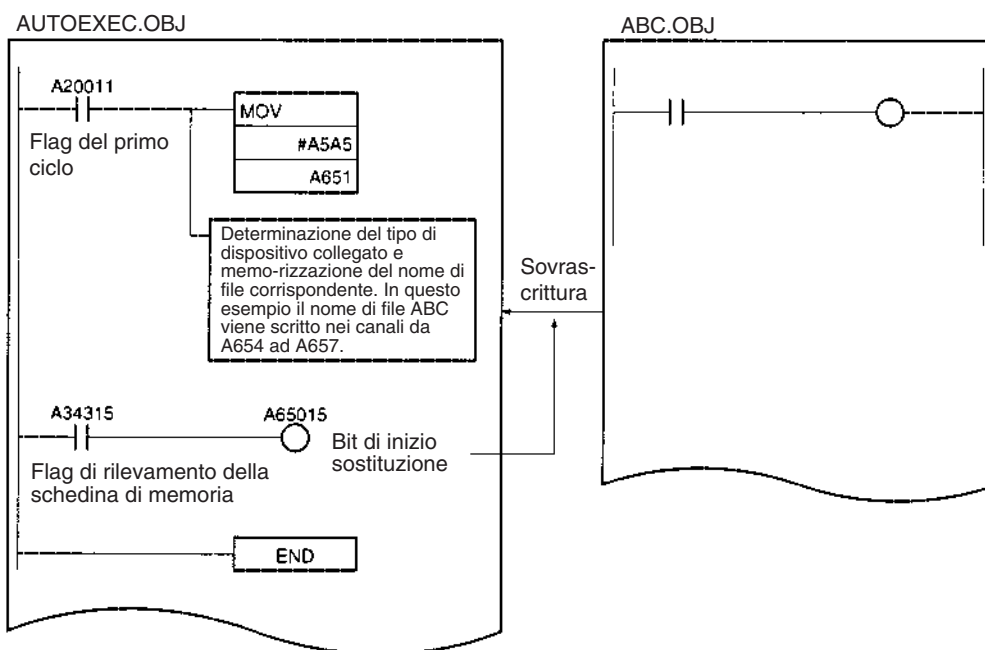
Nome	Indirizzo	Funzionamento
Bit di ritenzione dello stato forzato	A50013	Quando il bit è attivato, lo stato dei bit impostati o ripristinati in modo forzato viene mantenuto durante la sostituzione del programma.
Codice di completamento sostituzione (non supportato dalle CPU CS1 serie CS precedenti a EV1)	Da A65000 ad A65007	Codici di sostituzione normale del programma (A65014 OFF): 01 esadecimale: il file di programma (.OBJ) ha sostituito il programma. Codici di sostituzione incompleta del programma (A65014 ON): 00 esadecimale: si è verificato un errore fatale. 01 esadecimale: si è verificato un errore della memoria. 11 esadecimale: il programma è protetto da scrittura. 12 esadecimale: la password del programma in A651 non è corretta. 21 esadecimale: non è installata alcuna schedina di memoria. 22 esadecimale: il file specificato non esiste. 23 esadecimale: le dimensioni del file specificato sono troppo grandi (errore di memoria). 31 esadecimale: è in corso una delle seguenti operazioni: • È in corso un'operazione relativa alla memoria file. • È in corso la scrittura del programma. • È in corso la modifica della modalità operativa.
Flag di errore di sostituzione (non supportato dalle CPU CS1 serie CS precedenti a EV1)	A65014	Viene attivato quando si verifica un errore durante il tentativo di sostituzione del programma dopo l'attivazione del bit di inizio sostituzione (A65015). Viene disattivato alla successiva attivazione del bit A65015.
Bit di inizio sostituzione (non supportato dalle CPU CS1 serie CS precedenti a EV1)	A65015	Se il bit è stato abilitato impostando la password del programma (A651) su A5A5 esadecimale, la sostituzione del programma verrà avviata all'attivazione di questo bit. Non attivare nuovamente il bit durante la sostituzione del programma. Il bit viene automaticamente disattivato al termine della sostituzione del programma (con esito positivo o con errori) o all'accensione. È possibile verificare lo stato del bit utilizzando un dispositivo di programmazione, un terminale programmabile o un computer host per determinare se la sostituzione del programma è stata completata o meno.
Password del programma (non supportata dalle CPU CS1 serie CS precedenti a EV1)	A651	Scrivere la password in questo canale per abilitare la sostituzione del programma. A5A5 esadecimale: abilita il bit di inizio sostituzione (A65015). Altro valore: disabilita il bit di inizio sostituzione (A65015). Il bit viene automaticamente disattivato al termine della sostituzione del programma (con esito positivo o con errori) o all'accensione.
Nome di file del programma (non supportato dalle CPU CS1 serie CS precedenti a EV1)	Da A654 ad A657	Prima di avviare la sostituzione del programma, scrivere il nome del file del programma sostitutivo in formato ASCII in questi canali. Specificare solo gli 8 caratteri del nome senza l'estensione OBJ, in quanto verrà aggiunta automaticamente, a partire dal canale A654 (prima il byte più significativo). Se il nome di file contiene meno di 8 caratteri, inserire degli spazi nei byte rimanenti (20 esadecimale). Il nome di file non può contenere caratteri NULL o spazi. L'esempio seguente illustra i dati per il file di programma ABC.OBJ. 15 … 0 A654: 41 | 42 A655: 43 | 20 A656: 20 | 20 A657: 20 | 20

Programma di esempio 1

Memorizzare i file di programma ABC.OBJ e XYZ.OBJ nella schedina di memoria e selezionare il programma appropriato a seconda del valore di D00000. Impostare D00000 su #1234 quando si seleziona ABC.OBJ o su #5678 quando si seleziona XYZ.OBJ.

Avviare ed eseguire un altro task per le eventuali elaborazioni necessarie prima della sostituzione del programma o per l'elaborazione del bit di ritenzione IOM.

Programma di esempio 2

Memorizzare i file di programma per più dispositivi e il file di programma per il trasferimento automatico all'avvio, AUTOEXEC.OBJ o REPLACE.OBJ (vedere nota), in una schedina di memoria. Quando si accende il PLC, il file per il trasferimento automatico all'avvio viene letto e il programma viene successivamente sostituito con un file di programma per un altro dispositivo.

Nota Il file REPLACE.OBJ è supportato solo dalla CPU della serie CS/CJ versione 2.0 o successiva.

5-2-5 Trasferimento automatico all'avvio

È possibile utilizzare la funzione di trasferimento automatico all'avvio per trasferire il programma utente, i parametri e i dati della memoria I/O da una schedina di memoria alla CPU all'accensione del sistema.
I file riportati di seguito vengono letti e scritti nella memoria della CPU in modo automatico.

Nota Non è possibile utilizzare questa funzione per leggere la memoria file nell'area EM.

Il nome del file di programma varia a seconda che venga trasferito anche un file dell'area dei parametri.

Trasferimento di un file dell'area dei parametri

Utilizzare i nomi di file riportati di seguito.

File di programma: AUTOEXEC.OBJ
File dell'area dei parametri: AUTOEXEC.STD
File di dati: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM

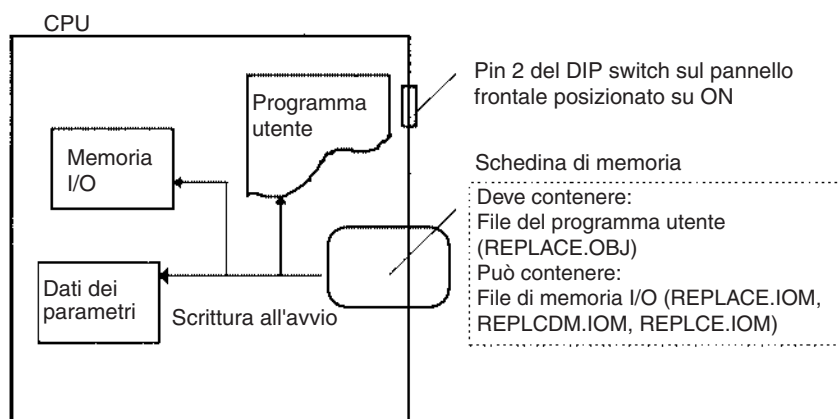
File	Nome file	Operazioni eseguite all'avvio	Requisito per il trasferimento automatico
File di programma	AUTOEXEC.OBJ	Il contenuto di questo file viene trasferito automaticamente e va a sovrascrivere l'intero programma utente, compresi gli attributi di task della CPU.	Deve trovarsi sulla schedina di memoria.
File dell'area dei parametri	AUTOEXEC.STD	Il contenuto di questo file viene trasferito automaticamente e va a sovrascrivere tutti i dati di configurazione iniziali della CPU.	Deve trovarsi sulla schedina di memoria.
File di dati	AUTOEXEC.IOM	Contiene i canali dell'area DM allocati a Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS). Il contenuto di questo file viene automaticamente trasferito all'area DM a partire dall'indirizzo D20000 all'accensione del sistema (vedere nota 1)	Non deve necessariamente trovarsi sulla schedina di memoria.
	ATEXECDM.IOM	Contiene canali DM di uso generico. Il contenuto di questo file viene automaticamente trasferito all'area DM a partire dall'indirizzo D00000 all'accensione del sistema (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1, vedere nota 1).	
	ATEXECE□.IOM	Contiene canali DM di uso generico. Il contenuto di questo file viene automaticamente trasferito all'area EM a partire da E□_00000 all'accensione del sistema (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1).	

Nota 1. Se i dati contenuti in AUTOEXEC.IOM e ATEXECDM.IOM si sovrappongono, i dati di ATEXECDM.IOM verranno sovrascritti ai dati in sovrapposizione trasferiti da AUTOEXEC.IOM, in quanto la scrittura di ATEXECDM.IOM è successiva.

2. Il file di programma (AUTOEXEC.OBJ) e il file dei parametri (AUTOEXEC.STD) devono trovarsi sulla schedina di memoria. In assenza di tali file, il trasferimento automatico non avrà esito positivo, si verificherà un errore di memoria e il flag di errore di memoria (A40115: errore fatale) verrà attivato. Viceversa, non è necessario che il file di dati relativo alla memoria I/O (AUTOEXEC.IOM) sia presente sulla schedina di memoria.
3. È possibile creare i file AUTOEXEC.IOM, ATEXECDM.IOM e ATEXECE□.IOM utilizzando un dispositivo di programmazione (Console di programmazione o CX-Programmer) e specificando, rispettivamente, indirizzi iniziali diversi da D20000, D00000 ed E□_00000. Poiché i dati verranno in ogni caso scritti a partire dall'indirizzo iniziale corretto, si consiglia di non specificare altri indirizzi iniziali.
4. Se il pin 7 del DIP switch viene impostato su ON e il pin 8 viene impostato su OFF per utilizzare la funzione di backup semplice, tale funzione avrà la precedenza anche se il pin 2 è impostato su ON. In tal caso, nella CPU non verranno trasferiti i file per il trasferimento automatico all'avvio bensì i file BACKUP□□ (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1).
5. La funzione di trasferimento automatico all'avvio può essere utilizzata in associazione con la funzione di sostituzione del programma. È possibile attivare il bit di inizio sostituzione (A65015) dal programma che viene trasferito automaticamente all'avvio per sostituirlo con un altro programma.
6. È possibile utilizzare contemporaneamente la funzione di trasferimento automatico all'avvio e la sostituzione completa del programma utilizzando i bit dell'area ausiliaria, ovvero il programma trasferito automaticamente alla CPU all'avvio può contenere la programmazione per manipolare i bit dell'area ausiliaria in modo da sostituire il programma con un altro.

Trasferimento senza un file dell'area dei parametri (solo CPU della serie CS/CJ versione 2.0 o successiva)

Utilizzare i nomi di file riportati di seguito.

File di programma: REPLACE.OBJ
File dell'area dei parametri: non richiesto e non trasferito indipendentemente dal nome del file.
File di dati: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM

File	Nome file	Operazioni eseguite all'avvio	Requisito per il trasferimento automatico
File di programma	REPLACE.OBJ Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	Il contenuto di questo file viene trasferito automaticamente e va a sovrascrivere l'intero programma utente, compresi gli attributi di task della CPU.	Deve trovarsi sulla schedina di memoria.
File dell'area dei parametri		Non trasferito indipendentemente dal nome del file.	Non richiesta

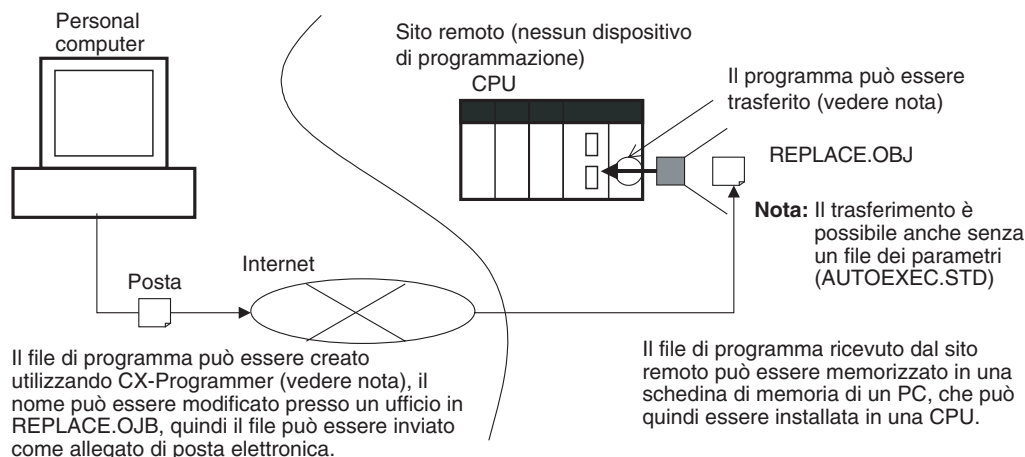
File	Nome file	Operazioni eseguite all'avvio	Requisito per il trasferimento automatico
File di dati	REPLACE.IOM Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	Contiene i canali dell'area DM allocati a Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS). Il contenuto di questo file viene automaticamente trasferito all'area DM a partire dall'indirizzo D20000 all'accensione del sistema	Non deve necessariamente trovarsi sulla schedina di memoria.
	REPLCDM.IOM Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	Contiene canali DM di uso generico. Il contenuto di questo file viene automaticamente trasferito all'area DM a partire dall'indirizzo D00000 all'accensione del sistema	
	REPLCE□.IOM Nota: solo CPU della serie CS/CJ versione 2.0 o successiva.	Contiene canali DM di uso generico. Il contenuto di questo file viene automaticamente trasferito all'area EM a partire da E□_00000 all'accensione del sistema □ indica il numero di banco.	

Nota

1. Se il nome del file di programma è REPLACE.OBJ (solo CPU della serie CS/CJ versione 2.0 o successiva), il file dell'area dei parametri non verrà trasferito anche se si trova nella schedina di memoria e a prescindere dal nome del file dell'area dei parametri.
2. Quando si crea il file REPLACE.IOM, REPLCDM.IOM o REPLCE□.IOM utilizzando un dispositivo di programmazione (Console di programmazione o CX-Programmer), specificare sempre il primo indirizzo appropriato a seconda dell'area dati (D20000, D00000 o E□_00000). Il contenuto del file verrà sempre trasferito a partire dal primo indirizzo appropriato a seconda dell'area dati (D20000, D00000 o E□_00000) anche se si specifica un indirizzo iniziale diverso, nel qual caso il contenuto di tale parte dell'area DM o del banco EM viene sovrascritto da dati errati.
3. Se il pin 7 del DIP switch viene impostato su ON e il pin 8 viene impostato su OFF per utilizzare la funzione di backup semplice, tale funzione avrà la precedenza anche se il pin 2 è impostato su ON. In tal caso, nella CPU non verranno trasferiti i file per il trasferimento automatico all'avvio bensì i file BACKUP□□

Esempio di applicazione

Trasferimento automatico dei file senza un file dell'area dei parametri

Un file di programma (.OBJ) può essere creato in modalità non in linea in un ufficio e trasferito in una postazione remota senza utilizzare alcun file dell'area dei parametri (.STD). È possibile memorizzare il file di programma in una schedina di memoria su un sito remoto senza utilizzare un dispositivo di programmazione ed è possibile utilizzare la schedina di memoria per il trasferimento automatico del programma alla CPU all'avvio.

Combinazioni di trasferimenti di file supportate

Nelle seguenti tabelle viene indicato se i file vengono trasferiti automaticamente alla CPU all'avvio, a seconda dei file presenti nella schedina di memoria.

■ File di programma: AUTOEXEC.OBJ

File di programma	File dell'area dei parametri	File di dati	Trasferito/ Non trasferito
AUTOEXEC.OBJ	AUTOEXEC.STD	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Trasferito
		Nessuno	
	Nessuno	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Non trasferito
		Nessuno	

■ File di programma: REPLACE.OBJ

File di programma	File dell'area dei parametri	File di dati	Trasferito/ Non trasferito
REPLACE.OBJ	Presente	Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Trasferito, ma il file dell'area dei parametri non viene trasferito
		Nessuno	
	Nessuno	Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Trasferito
		Nessuno	

■ Nessun file di programma

File di programma	File dell'area dei parametri	File di dati	Trasferito/ Non trasferito
Nessuno	AUTOEXEC.STD	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Non trasferito
		Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
		Nessuno	
	Nessuno	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	
		Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	

■ File AUTOEXEC e REPLACE

<u>Più file di programma</u>

File di programma		File dell'area dei parametri	File di dati	Trasferito/ Non trasferito
AUTOEXEC. OBJ	REPLACE OBJ	AUTOEXEC.STD	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Non trasferito
			Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	
			Nessuno	
		Nessuno	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	
			Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	

Più di un tipo di file di dati

File di programma	File dell'area dei parametri	File di dati		Trasferito/Non trasferito
AUTOEXEC.OBJ	AUTOEXEC.STD	Uno o più dei seguenti file: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM	Uno o più dei seguenti file: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM	Vengono trasferiti i seguenti file di dati: AUTOEXEC.IOM, ATEXECDM.IOM, ATEXECE□.IOM
	Nessuno			Non trasferito
REPLACE.OBJ	Ignorato			Vengono trasferiti i seguenti file di dati: REPLACE.IOM, REPLCDM.IOM, REPLCE□.IOM

Procedura

1,2,3...

1. Spegnere il PLC.
2. Impostare su ON il pin 2 del DIP switch sul pannello frontale della CPU. Accertarsi che i pin 7 e 8 siano entrambi impostati su OFF.

   Nota Poiché la funzione di backup semplice avrà la precedenza sulla funzione di trasferimento automatico all'avvio, accertarsi che i pin 7 e 8 siano impostati su OFF.
3. Preparare una schedina di memoria come descritto di seguito.
   a) Trasferimento con un file dell'area dei parametri
   Inserire una schedina di memoria con il file del programma utente (AUTOEXEC.OBJ), il file dell'area dei parametri (AUTOEXEC.STD) e/o i file di dati relativi alla memoria I/O (AUTOEXEC.IOM, ATEXECDM.IOM e ATEXECE□.IOM) creati con CX-Programmer. È necessario che sulla schedina di memoria siano presenti i file del programma e dell'area dei parametri. I file di dati della memoria I/O sono opzionali.
   b) Trasferimento senza un file dell'area dei parametri
   Inserire una schedina di memoria con il file del programma utente (REPLACE.OBJ) e/o i file di dati relativi alla memoria I/O (REPLACE.IOM, REPLCDM.IOM e REPLCE□.IOM) creati con CX-Programmer. Nella schedina di memoria deve essere presente il file del programma, mentre i file della memoria I/O sono opzionali.
4. Accendere il PLC.

Nota **Errore di trasferimento automatico all'avvio**
Se il trasferimento automatico all'avvio non ha esito positivo, si verificherà un errore di memoria, verrà attivato il flag A40115 e la CPU si arresterà. In caso di errore, spegnere il sistema per cancellare la condizione di errore. La condizione di errore può essere cancellata esclusivamente interrompendo l'alimentazione.

DIP switch sul pannello frontale della CPU

Pin	Nome	Impostazione
2	Pin della funzione di trasferimento automatico all'avvio	ON: viene eseguito il trasferimento automatico all'avvio. OFF: non viene eseguito il trasferimento automatico all'avvio.
7 e 8	Pin della funzione di backup semplice	Entrambi impostati su OFF.

Bit e canali ausiliari correlati

Nome	Indirizzo	Impostazione
Flag di errore di memoria (errore fatale)	A40115	Viene attivato quando all'accensione si verifica un errore di memoria o si verifica un errore durante il trasferimento automatico dalla schedina di memoria (trasferimento automatico all'avvio). La CPU si arresta e l'indicatore ERR/ALM sulla parte frontale della CPU si accende. **Nota:** se l'errore si verifica durante il trasferimento automatico all'avvio, viene attivato il flag A40309. In questo caso non è possibile cancellare la condizione di errore.
Flag di errore di trasferimento all'avvio della schedina di memoria	A40309	Viene attivato quando il trasferimento automatico all'avvio risulta selezionato (pin 2 del DIP switch impostato su ON) e si verifica un errore durante l'operazione. L'errore si verifica quando il file specificato non esiste, la schedina di memoria non è installata o si verifica un problema durante il trasferimento. **Nota:** la condizione di errore può essere cancellata solo spegnendo il Modulo e non in presenza di alimentazione.

Precauzioni per la modifica dello stato di allocazione degli I/O durante il trasferimento automatico all'avvio

Lo stato di allocazione degli I/O dipende dalle versioni delle CPU di origine e di destinazione quando si utilizza una CPU della serie CJ per creare i file dei parametri per il trasferimento automatico all'avvio, per salvarli in una memory card e quindi per trasferirli automaticamente a un'altra CPU della serie CJ all'avvio. Le modifiche allo stato di allocazione degli I/O per le combinazioni di diverse versioni della CPU sono riportate nella seguente tabella.

CPU di origine		Stato di allocazione degli I/O di origine	CPU di destinazione		
			Versione della CPU a cui verranno inviati i file per il trasferimento automatico all'avvio		
			CPU precedenti alla versione 2.0	**CPU versione 2.0**	**CPU versione 3.0 o successiva**
Versione della CPU utilizzata per creare i file per il trasferimento automatico all'avvio	CPU precedenti alla versione 2.0	Allocazione automatica	Passa a quella definita dall'utente	(Uguale) Allocazione automatica	Passa a quella definita dall'utente
		Definita dall'utente	(Uguale) Definita dall'utente	Passa all'allocazione automatica (vedere nota 1)	(Uguale) Definita dall'utente
	CPU versione 2.0	Allocazione automatica	Passa a quella definita dall'utente	(Uguale) Allocazione automatica	Passa al funzionamento utente
		Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente
	Numero versione modulo 3.0 o successiva	Allocazione automatica	Passa a quella definita dall'utente	(Uguale) Allocazione automatica	(Uguale) Allocazione automatica
		Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente	(Uguale) Definita dall'utente

Nota

1. Se i file per il trasferimento automatico all'avvio (AUTOEXEC.STD) vengono creati e salvati in una memory card utilizzando le allocazioni degli I/O definite dall'utente con una CPU della serie CJ precedente alla versione 2.0 e i dati vengono trasferiti automaticamente da una memory card, all'avvio il sistema passa automaticamente all'allocazione degli I/O automatica.
2. Se i file per il trasferimento automatico all'avvio (AUTOEXEC.STD) vengono creati e salvati in una memory card utilizzando una CPU della serie CJ versione 2.0 e i dati vengono trasferiti automaticamente da una memory card a una CPU della serie CJ versione 3.0 o successiva, il sistema passa automaticamente all'allocazione degli I/O definita dall'utente.

Precauzioni per il confronto di file dei parametri di trasferimento automatico all'avvio

È possibile che sulla Console di programmazione si verifichino degli errori quando si confrontano dati dei parametri tra file prima e dopo il trasferimento dei dati se si creano file dei parametri di trasferimento automatico all'avvio (AUTOEXEC.STD) e si esegue il trasferimento automatico all'avvio per combinazioni di CPU della serie CJ precedenti alla versione 2.0, versione 2.0 e versione 3.0.

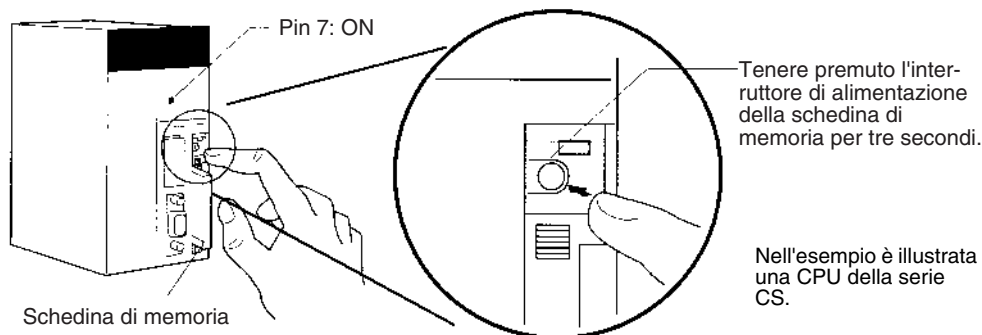
CPU di origine		Stato di allocazione degli I/O di origine	CPU di destinazione		
			Versione della CPU alla destinazione di backup/ripristino		
			CPU precedenti alla versione 2.0	CPU versione 2.0 o successiva	CPU versione 3.0 o successiva
Versione della CPU all'origine di backup	CPU precedenti alla versione 2.0	Allocazione automatica	Verifica possibile	Verifica possibile	Errore di verifica
		Definita dall'utente			Errore di verifica
	CPU versione 2.0 o successiva	Allocazione automatica			Errore di verifica
		Definita dall'utente			Verifica possibile
	CPU versione 3.0 o successiva	Allocazione automatica		Errore di verifica	Verifica possibile
		Definita dall'utente		Verifica possibile	

5-2-6 Funzione di backup semplice

Questa funzione non è supportata dalle CPU CS1 della serie CS precedenti a EV1.

Backup dei dati dalla CPU alla memory card

Per eseguire il backup dei dati, impostare il pin 7 del DIP switch della CPU su ON e tenere premuto l'interruttore di alimentazione della memory card per tre secondi. La funzione di backup crea automaticamente file di backup con estensioni e nomi di file fissi e li scrive nella memory card. I file di backup contengono il programma, i dati dell'area dei parametri, i dati della memoria I/O, le tabelle dei simboli (vedere nota), i file di commento (vedere nota) e i file di indice dei programmi (vedere nota). È possibile eseguire questa funzione in qualsiasi modalità operativa.

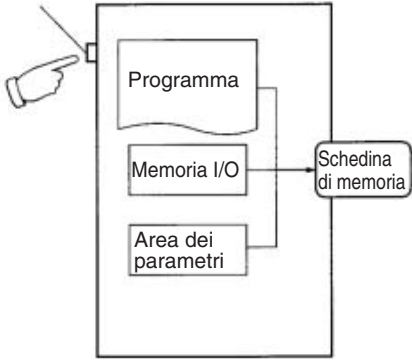
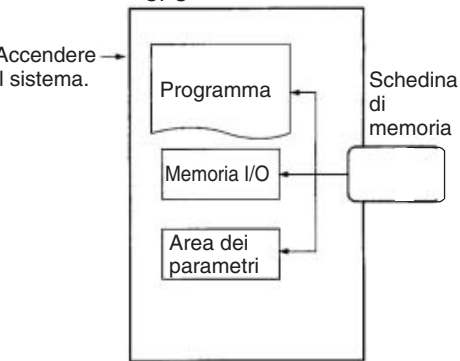
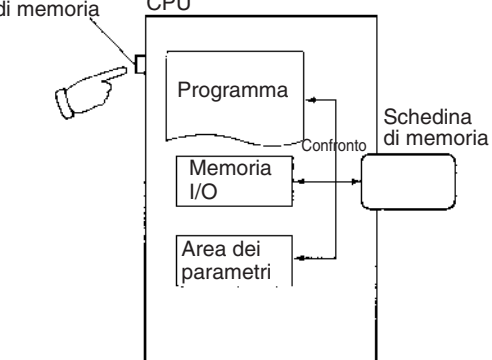
Nota Questi dati sono supportati dalle CPU della serie CS/CJ versione 3.0 o successiva. I file di backup vengono creati automaticamente dai file nella memory card, nella memoria file nell'area EM o nella memoria dei commenti.

Ripristino dei dati dalla schedina di memoria alla CPU

Per ripristinare i file di backup sulla CPU, verificare che il pin 7 sia impostato su ON, quindi spegnere e riaccendere il PLC. I file di backup contenenti il programma, i dati dell'area dei parametri e i dati della memoria I/O verranno trasferiti dalla schedina di memoria alla CPU.

*1 Questi dati sono supportati dalle CPU della serie CS/CJ versione 3.0 o successiva. I file di backup vengono letti nella memory card, nella memoria file nell'area EM o nella memoria dei commenti.

Nota

1. Poiché la funzione di backup ha la precedenza sulla funzione di trasferimento automatico all'avvio, i file di backup verranno trasferiti sulla CPU all'accensione del PLC anche se il pin 2 del DIP switch è impostato su ON.
2. Se il pin 1 del DIP switch è impostato su ON (protezione da scrittura della memoria di programma), i dati non verranno trasferiti dalla schedina di memoria alla CPU.
3. Quando i file di backup vengono letti dalla schedina di memoria tramite la funzione di backup, lo stato della memoria I/O e dei bit impostati o ripristinati in modo forzato verrà cancellato, a meno che non siano state configurate le impostazioni necessarie nell'area ausiliaria e nella configurazione del PLC.

   Se il bit di ritenzione IOM (A50012) è attivato e la configurazione del PLC è impostata in modo da mantenere lo stato di tale bit all'avvio durante la scrittura dei file di backup, lo stato dei dati della memoria I/O verrà mantenuto durante la lettura dei dati dalla schedina di memoria.

   Se il bit di ritenzione dello stato forzato (A50013) è attivato e la configurazione del PLC è impostata in modo da mantenere lo stato di tale bit all'avvio durante la scrittura dei file di backup, lo stato dei bit impostati o ripristinati in modo forzato verrà mantenuto durante la lettura dei dati dalla schedina di memoria.
4. Dopo l'esecuzione dell'operazione di backup semplice, le CPU CS1-H, CJ1-H, CJ1M o CS1D restano in modalità PROGRAM e non sarà possibile passare alla modalità MONITOR o RUN finché le CPU non verranno spente e riaccese. Al termine dell'operazione di backup, spegnere la CPU, modificare l'impostazione del pin 7, quindi riaccendere la CPU.
5. Le operazioni di backup dei file possono richiedere da diversi secondi a diversi minuti. Per informazioni sui tempi di esecuzione, fare riferimento a pagina 254.

Confronto tra i dati della schedina di memoria e della CPU

Per eseguire il confronto tra i file di backup della memory card e i dati della CPU, impostare il pin 7 del DIP switch della CPU su OFF, quindi tenere premuto l'interruttore di alimentazione della memory card per tre secondi. La funzione di backup confronterà il programma, i dati dell'area dei parametri, i dati della memoria I/O, le tabelle dei simboli (vedere nota), i file di commento (vedere nota) e i file di indice dei programmi (vedere nota) nella memory card con i dati corrispondenti nella CPU. È possibile eseguire questa funzione in qualsiasi modalità operativa.

Nota Questi dati sono supportati dalle CPU della serie CS/CJ versione 3.0 o successiva.

Nella tabella riportata di seguito sono sintetizzate le operazioni della funzione di backup semplice.

Operazione di backup	Stato del pin	Procedura
	Pin 7	
Backup dei dati dalla CPU alla schedina di memoria Backup dei dati sulla schedina di memoria Interruttore di alimentazione della schedina di memoria CPU Programma Memoria I/O Area dei parametri Schedina di memoria Pin 7: ON	ON	Tenere premuto l'interruttore di alimentazione della schedina di memoria per tre secondi.
Ripristino dei dati dalla schedina di memoria alla CPU Ripristino dei dati dalla schedina di memoria CPU Accendere il sistema. Programma Memoria I/O Area dei parametri Schedina di memoria Pin 7: ON	ON	Spegnere e riaccendere il PLC (vedere nota 1)
Confronto tra i dati della schedina di memoria e della CPU Interruttore di alimentazione della schedina di memoria Confronto dei dati con quelli sulla schedina di memoria CPU Programma Confronto Memoria I/O Area dei parametri Schedina di memoria Pin 7: OFF	OFF	Tenere premuto l'interruttore di alimentazione della schedina di memoria per tre secondi.

Nota

1. Per ulteriori informazioni sui risultati delle operazioni di lettura, scrittura e confronto, fare riferimento alla sezione *Verifica delle operazioni di backup tramite indicatori* a pagina 242.
2. Per indicazioni sul tempo richiesto per le operazioni di backup sulla schedina di memoria, fare riferimento alla sezione *5-3-2 Procedure operative*.

File di backup

File di dati

Nome file ed estensione	Area dati e intervallo di indirizzi memorizzato		Backup dalla memoria I/O alla schedina di memoria (creazione dei file)	Ripristino dalla memory card alla CPU	Confronto tra memory card e CPU		File necessari per il ripristino dei dati
CPU	**CS/CJ**				**CS1/CJ1**	**CS1-H/CJ1-H**	
BACKUP.IOM	DM	Da D20000 a D32767	Sì	Sì	Sì	---	Deve trovarsi sulla schedina di memoria.
BACKUPIO.IOR	CIO	Da 0000 a 6143 (compreso stato dei bit forzato)	Sì	---[4]	Sì	---	Deve trovarsi sulla schedina di memoria.
	WR	Da W000 a W511 (compreso stato dei bit forzato)	Sì	---[4]	Sì	---	
	HR	Da H000 a H511	Sì	Sì	Sì	---	
	AR	Da A000 ad A447	Sì	---	---	---	
		Da A448 ad A959	Sì	Sì	Sì	---	
	Temporizzatore[1]	Da T0000 a T4095	Sì	Sì[4]	Sì	---	
	Contatore[1]	Da C0000 a C4095	Sì	Sì	Sì	---	
BACKUPDM.IOM	DM	Da D00000 a D19999	Sì	Sì	Sì	---	Deve trovarsi sulla schedina di memoria.
BACKUPE□.IOM[2, 3]	EM	Da E□_00000 a E□_32767	Sì	Sì	Sì	---	Devono trovarsi sulla schedina di memoria (corrispondenza con i banchi della CPU).

Nota

1. I flag di completamento e i valori attuali vengono inclusi nel backup.
2. Il simbolo □ rappresenta il numero del banco, e il numero di banchi dipende dalla CPU in uso.

   Durante il ripristino sulla CPU, i file BACKUPE□.IOM presenti nella schedina di memoria vengono letti in ordine di sequenza, a partire dal banco 0 fino all'ultimo numero di banco disponibile nella CPU. Se il numero di banchi compresi nel backup è superiore al numero di banchi della CPU, i file BACKUPE□.IOM in eccesso non verranno letti. Viceversa, se il numero di banchi compresi nel backup è inferiore al numero di banchi disponibili nella CPU, i banchi rimanenti della CPU rimarranno invariati.

   Se un file BACKUPE□.IOM risulta mancante, ad esempio se è stato eseguito il backup solo dei banchi 0, 1, 2, 4, 5 e 6, verranno ripristinati solo i file consecutivi. In questo caso, vengono letti solo i dati per i banchi 0, 1 e 2.

3. I dati dell'area EM vengono inseriti nel backup come dati binari. I banchi EM convertiti in memoria file verranno inseriti nel backup insieme ai banchi EM non convertiti.

   È possibile ripristinare la memoria file nell'area EM di un'altra CPU solo se i file BACKUPE□.IOM sono consecutivi e il numero di banchi EM compresi nel backup corrisponde al numero di banchi della CPU. Se i file BACKUPE□.IOM non sono consecutivi o il numero di banchi EM non corrisponde al numero di banchi disponibili nella CPU, la porzione dell'area EM formattata come memoria file nell'area EM perderà tale formattazione e i dati relativi alla memoria file non saranno validi. I banchi dell'area EM corretti verranno letti normalmente.

4. Di norma, all'accensione del PLC, il contenuto dell'area CIO e dell'area WR, i flag di completamento del temporizzatore, i valori attuali del temporizzatore e lo stato dei bit impostati o ripristinati in modo forzato vengono cancellati e viene letto il file BACKUPIO.IOR dalla schedina di memoria.

   Se il bit di ritenzione IOM (A50012) è attivato e la configurazione del PLC è impostata in modo da mantenere lo stato di tale bit all'avvio durante la scrittura dei file di backup, lo stato dei dati della memoria I/O verrà mantenuto durante la lettura dei dati dalla schedina di memoria.

   Se il bit di ritenzione dello stato forzato (A50013) è attivato e la configurazione del PLC è impostata in modo da mantenere lo stato di tale bit all'avvio durante la scrittura dei file di backup, lo stato dei bit impostati o ripristinati in modo forzato verrà mantenuto durante la lettura dei dati dalla schedina di memoria.

File di programma

Nome file ed estensione	Contenuto	Backup dalla memoria I/O alla schedina di memoria (creazione dei file)	Ripristino dalla memory card alla CPU	Confronto tra memory card e CPU	File necessari per il ripristino dei dati
CPU	**CS/CJ**				
BACKUP.OBJ	Intero programma utente	Sì	Sì	Sì	Deve trovarsi sulla schedina di memoria.

File di parametri

Nome file ed estensione	Contenuto	Backup dalla memoria I/O alla schedina di memoria (creazione dei file)	Ripristino dalla memory card alla CPU	Confronto tra memory card e CPU	File necessari per il ripristino dei dati
CPU	**CS/CJ**				
BACKUP.STD	Configurazione del PLC Tabelle di I/O registrati Tabelle di routing Configurazione del Modulo CPU bus ecc.	Sì	Sì	Sì	Deve trovarsi sulla schedina di memoria.

File di backup di moduli e schede specifici Solo CPU CS1-H, CS1D, CJ1-H, CJ1M o CS1D

Nome file ed estensione	Contenuto	Backup dalla memoria I/O alla schedina di memoria (creazione dei file)	Ripristino dalla memory card alla CPU	Confronto tra memory card e CPU	File necessari per il ripristino dei dati
CPU	**Solo CPU CS1-H, CS1D, CJ1-H, CJ1M o CS1D**				
BACKUP□□.PRM (dove □□ è l'indirizzo del modulo o della scheda di cui si esegue il backup)	Backup dei dati del modulo o della scheda con l'indirizzo specificato (il contenuto specifico dipende dal modulo o dalla scheda)	Sì	Sì	Sì	Devono trovarsi sulla schedina di memoria (vedere nota 2).

Nota 1. Gli indirizzi di modulo sono:
Moduli CPU bus: numero modulo + 10 esadecimale
Moduli di I/O speciali: numero modulo + 20 esadecimale
Scheda interna: E1 esadecimale

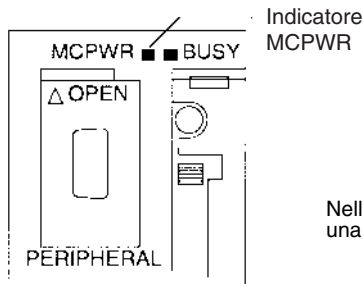
2. Anche se questi file non dovessero essere presenti sulla schedina di memoria al momento del trasferimento dei dati alla memoria I/O, non si verificheranno errori nella CPU. Tuttavia, si verificherà un errore nel modulo o nella scheda corrispondente qualora i dati non vengano ripristinati. Per ulteriori informazioni sugli errori relativi a specifici moduli e schede, consultare il manuale dell'operatore del modulo o della scheda in questione.

Tabelle dei simboli, file di commento e file di indice dei programmi (solo CPU (CS1-H/CJ1-H, CJ1M, CS1D versione 3.0 o successiva)

Nome file ed estensione	Contenuto	Backup dalla CPU alla memory card (creazione di file)	Ripristino dalla memory card alla CPU	Confronto tra memory card e CPU	File necessari per il ripristino dei dati
CPU	**Solo CPU CS1-H, CJ1-H, CJ1M o CS1D versione 3.0**				
BKUPSYM.SYM	File delle tabelle dei simboli	Sì (scrittura se la memoria dei commenti della CPU contiene file)	Sì (lettura nella memoria dei commenti della CPU se la memory card contiene file)	Sì (confronto con i file contenuti nella memoria dei commenti della CPU)	Questi file non devono essere necessariamente presenti nella memory card.
BKUPCMT.CMT	File di commento				
BKUPPRG.IDX	File di indice dei programmi				

Nota Le operazioni di backup, ripristino e confronto nella precedente tabella vengono eseguite solo per i file contenuti nella memoria dei commenti.

Verifica delle operazioni di backup tramite indicatori

Lo stato dell'indicatore di alimentazione della schedina di memoria (MCPWR) segnala se l'operazione di backup semplice è stata completata in modo corretto.

Nell'esempio è illustrata una CPU della serie CS.

Operazione di backup	Completamento corretto (vedere nota 1)	Presenza di errori	
	Stato indicatore MCPWR	**Stato indicatore MCPWR**	**errore**
Backup dei dati dalla CPU alla schedina di memoria	Acceso → Si illumina a rimane acceso mentre si tiene premuto l'interruttore di alimentazione della schedina di memoria. → Lampeggia una volta. → Rimane acceso durante la scrittura. → Si spegne al termine della scrittura dei dati.	Acceso → Si illumina a rimane acceso mentre si tiene premuto l'interruttore di alimentazione della schedina di memoria. → Continua a lampeggiare. → Si accende quando si preme l'interruttore di alimentazione della schedina di memoria.	Non verrà creato alcun file in presenza dei seguenti errori: Capacità insufficiente della schedina di memoria (vedere nota 2) Errore di memoria della CPU Errore del bus di I/O (durante la scrittura dei dati su un modulo o una scheda, solo CPU CS1-H, CS1D o CJ1-H)
Ripristino dei dati dalla schedina di memoria alla CPU	Si illumina all'accensione. → Lampeggia una volta. → Rimane acceso durante la lettura. → Si spegne al termine della lettura dei dati.	Si illumina all'accensione. → Lampeggia cinque volte. → Si spegne.	I dati non verranno letti in presenza dei seguenti errori: Il programma sulla schedina di memoria supera la capacità della CPU. I file di backup necessari non sono presenti sulla schedina di memoria. Non è possibile scrivere il programma perché è protetto da scrittura (pin 1 del DIP switch impostato su ON).
		Si illumina all'accensione. → Lampeggia una volta. → Rimane acceso durante la lettura. → Lampeggia tre volte. → Si spegne al termine della lettura dei dati.	Attenzione: i dati verranno letti in presenza del seguente errore. I file EM e i banchi EM della CPU non corrispondono, ossia i numeri di banco non sono consecutivi o il numero di banchi non corrisponde.

Operazione di backup	Completamento corretto (vedere nota 1)	Presenza di errori	
	Stato indicatore MCPWR	Stato indicatore MCPWR	errore
Confronto tra i dati della schedina di memoria e della CPU	Acceso → Si illumina a rimane acceso mentre si tiene premuto l'interruttore di alimentazione della schedina di memoria. → Lampeggia una volta. → Rimane acceso durante il confronto. → Si spegne al termine del confronto dei dati.	Acceso → Si illumina a rimane acceso mentre si tiene premuto l'interruttore di alimentazione della schedina di memoria. → Continua a lampeggiare. → Si accende quando si preme l'interruttore di alimentazione della schedina di memoria.	È possibile che si siano verificati i seguenti errori di confronto (vedere nota 3): I dati della schedina di memoria e della CPU non corrispondono. I file di backup necessari non sono presenti sulla schedina di memoria. I file EM e i banchi EM della CPU non corrispondono, ossia i numeri di banco non sono consecutivi o il numero di banchi non corrisponde. Errore di memoria della CPU Errore del bus di I/O (durante il confronto dei dati su un modulo o una scheda, solo CPU CS1-H, CS1D o CJ1-H)
Comune alle tre operazioni di backup	---	Lettura: lampeggia cinque volte. → Si spegne. Scrittura o confronto: continua a lampeggiare.→ si accende quando si preme l'interruttore di alimentazione della schedina di memoria.	Errore di accesso alla schedina di memoria (errore di formattazione o di lettura/scrittura)

Nota

1. Quando l'operazione di backup viene completata in modo corretto, lo spegnimento dell'indicatore MCPWR indica che la schedina di memoria non è più alimentata. Per utilizzare nuovamente la schedina di memoria, premere l'interruttore di alimentazione della schedina per ripristinare l'alimentazione ed eseguire l'operazione desiderata.
2. Quando i dati vengono scritti per un'operazione di backup semplice da una CPU CS1-H, CS1D, CJ1-H, CJ1M o CS1D, è possibile verificare gli errori dovuti a capacità insufficiente della schedina di memoria in A397 (capacità di scrittura per backup semplice). Se, dopo l'operazione di scrittura, A397 contiene un valore diverso da 0000 esadecimale, tale valore indicherà la capacità in KB richiesta sulla schedina di memoria.
3. Per le CPU CS1-H, CS1D, CJ1-H, CJ1M o CS1D, vengono confrontati anche i file di backup di moduli e schede.

Bit e canali ausiliari correlati

Nome	Indirizzo	Descrizione
Flag di utilizzo della memoria file	A34313	Viene attivato quando è in esecuzione una delle operazioni elencate di seguito e viene disattivato al termine dell'esecuzione. • Rilevamento della schedina di memoria • Esecuzione dell'istruzione CMND per la CPU locale • Esecuzione delle istruzioni FREAD e FWRIT • Sostituzione del programma tramite bit di controllo speciali • Operazione di backup semplice Il trasferimento dei dati e la verifica del contenuto della schedina di memoria non sono possibili quando il flag è attivato.
Banco di inizio della memoria file nell'area EM	A344	Quando la CPU inizia la lettura dei dati della schedina di memoria, fa riferimento a questo valore. Se il numero massimo di banchi EM dei file BACKUPE□.IOM (numero massimo di banchi consecutivi a partire da 0) corrisponde al numero massimo di banchi della CPU, l'area EM verrà formattata in base al valore di questo canale. Se il numero massimo di banchi EM non corrisponde, l'area EM tornerà alla condizione non formattata.
Flag di abilitazione istruzioni di comunicazione in rete (solo CPU CS1-H, CJ1-H, CJ1M o CS1D, vedere nota)	Da A20200 ad A20207	• Viene disattivato quando ha inizio la scrittura o il confronto dei dati della schedina di memoria. • Viene attivato al termine della scrittura o del confronto dei dati della schedina di memoria. Se tutti i flag di abilitazione delle istruzioni di comunicazione in rete sono disattivati quando vengono avviate le operazioni di scrittura o confronto della schedina di memoria, non sarà possibile scrivere o confrontare i dati relativi ai moduli e alle schede. Se si tenta di eseguire questa operazione, si verificherà un errore.
Codice di completamento comunicazione in rete (solo CPU CS1-H, CJ1-H, CJ1M o CS1D, vedere nota)	Da A203 ad A210	Indica i risultati della comunicazione con il modulo o la scheda quando vengono eseguite operazioni di scrittura o confronto della schedina di memoria.
Flag di errore di comunicazione in rete (solo CPU CS1-H, CJ1-H, CJ1M o CS1D, vedere nota)	Da A21900 ad A21907	• Viene attivato se si verifica un errore di comunicazione con il modulo o la scheda quando vengono eseguite operazioni di scrittura o confronto della schedina di memoria. • Rimane disattivato (o viene disattivato) se non si verifica alcun errore di comunicazione con il modulo o la scheda quando vengono eseguite operazioni di scrittura o confronto della schedina di memoria.
Capacità di scrittura per backup semplice (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)	A397	Segnala un errore di memoria per capacità insufficiente e indica la capacità in KB richiesta per i dati sulla schedina di memoria quando la scrittura per un'operazione di backup semplice non ha esito positivo. Da 0001 a FFFF esadecimale: errore di scrittura (indica la capacità richiesta sulla schedina di memoria, compresa tra 1 e 65.535 KB). Viene cancellato e impostato sul valore 0000 esadecimale quando la scrittura viene portata a termine correttamente. 0000 esadecimale: scrittura completata correttamente.

Nota Questi flag sono correlati alle CPU CS1-H, CJ1-H, CJ1M o CS1D in quanto tali CPU utilizzano automaticamente una porta di comunicazione disponibile per la scrittura o il confronto dei dati di una schedina di memoria.

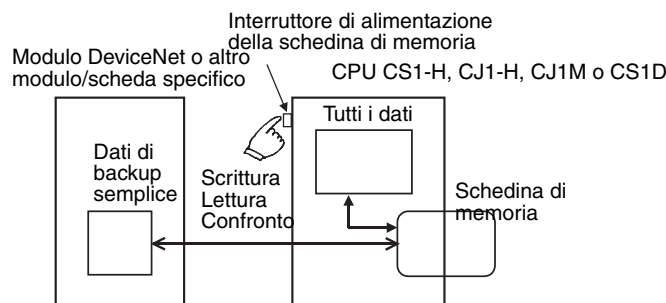
Backup dei dati di moduli e schede

Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Introduzione

Durante l'operazione di backup semplice, viene eseguito il backup dei seguenti dati della CPU (CS1 e CJ1): programma utente, area dei parametri, intera memoria I/O. Oltre a tali dati, per le CPU CS1-H, CJ1-H, CJ1M o CS1D viene eseguito anche il backup dei dati relativi a moduli e schede specifici installati sul PLC.

Descrizione del funzionamento

Quando si esegue un'operazione di backup semplice per una CPU CS1-H, CJ1-H, CJ1M o CS1D, sulla schedina di memoria viene memorizzato anche un file di backup che contiene dati relativi a moduli e schede specifici. I dati vengono memorizzati separatamente per ciascun modulo o scheda.

Ambito di applicazione

È possibile utilizzare questa funzione per eseguire il backup dei dati per l'intero PLC, compresi la CPU, i Moduli DeviceNet, Moduli e Schede di comunicazione seriale e così via. È inoltre possibile utilizzare tale funzione in caso di sostituzione di un modulo.

File di backup di moduli e schede specifici

I dati di ciascun modulo o scheda vengono memorizzati nella schedina di memoria utilizzando il nome file BACKUP□□.PRM, dove "□□" è l'indirizzo del modulo o della scheda espresso in valori esadecimali.

Nota Gli indirizzi di modulo sono:
Moduli CPU bus: numero modulo + 10 esadecimale
Moduli di I/O speciale: numero modulo + 20 esadecimale
Scheda interna: E1 esadecimale

Questi file vengono inoltre utilizzati per la lettura dalla schedina di memoria o il confronto dei dati della schedina di memoria.

Moduli e schede applicabili

È necessario che i moduli e le schede di cui si desidera memorizzare i dati supportino la funzione di backup. Per ulteriori informazioni sul supporto di tale funzione, consultare il manuale dell'operatore del modulo o della scheda in questione.

Modulo/ scheda	Codici di modello	Dati di cui viene eseguito il backup semplice quando utilizzati con la CPU CS1-H/CJ1-H	Capacità dei dati utilizzata nella schedina di memoria per il backup semplice
Moduli DeviceNet	CJ1W-DRM21-V1 CJ1W-DRM21	Parametri del dispositivo (tutti i dati nella EEPROM del Modulo) Sebbene si tratti dei medesimi dati memorizzati dalla funzione di backup della schedina di memoria supportata dal Modulo o dalla configurazione DeviceNet (versione 2.0), i file non sono compatibili.	7 Kb
Moduli di comunicazione seriale	CS1W-SCU21-V1 CJ1W-SCU41	Dati delle protocol macro Sono inclusi sia i protocolli di sistema standard che i protocolli definiti dall'utente memorizzati nella memoria flash del Modulo o della Scheda.	129 Kb
Schede di comunicazione seriale	CS1W-SCB21-V1 CS1W-SCB41-V1		129 Kb

Modulo/ scheda	Codici di modello	Dati di cui viene eseguito il backup semplice quando utilizzati con la CPU CS1-H/CJ1-H	Capacità dei dati utilizzata nella schedina di memoria per il backup semplice
Contatori personalizzabili	CS1W-HIO01-V1 CS1W-HCP22-V1 CS1W-HCA22-V1 CS1W-HCA12-V1	• Programma utente • Area DM di sola lettura di uso generico • Area di impostazione delle funzioni dei moduli • Informazioni sulle istruzioni di espansione • Libreria programmazione ladder	64 Kb
Moduli di posizionamento	CS1W-MCH71	• Dati di posizionamento • Parametri di sistema • Programmi in linguaggio G	8.192 Kb
	CS1W-MC221-V1 CS1W-MC421-V1		142 Kb
Moduli di posizionamento	CS1W-NC113/133/213/233/413/433 versione 2.0 o successiva CJ1W-NC113/133/213/233/413/433 versione 2.0 o successiva	• Parametri degli assi • Dati sequenze • Dati velocità • Dati tempo di accelerazione/decelerazione • Dati temporizzatore doppio • Dati di zona	7 Kb

Nota I dati dei Moduli e delle Schede sopra elencati vengono automaticamente inclusi nel backup durante l'operazione di backup semplice. Non esistono impostazioni per l'inclusione o esclusione di tali dati. Se si utilizza una Console di programmazione, tuttavia, le operazioni sono supportate singolarmente per l'area del programma utente, l'area dei parametri e le aree della memoria I/O. Per ulteriori informazioni, fare riferimento al manuale *Programming Console Operation Manual* (W314).

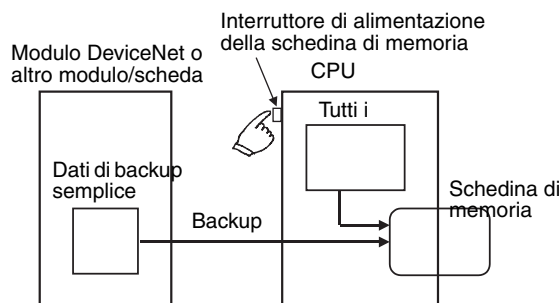
Procedura

La procedura per l'operazione di backup semplice non varia, indipendentemente dal fatto che venga eseguito o meno il backup dei dati relativi a moduli e schede specifici (scrittura, lettura e confronto inclusi).

■ **Backup dei dati**

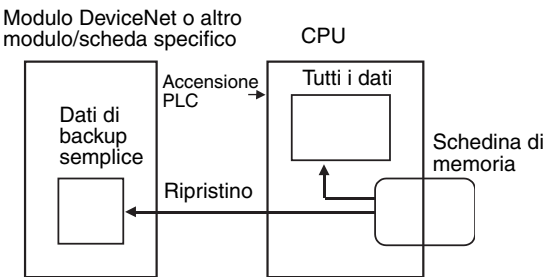
1,2,3...

1. Posizionare su ON il pin 7 del DIP switch della CPU.
2. Tenere premuto l'interruttore di alimentazione della schedina di memoria per tre secondi.

   I dati di backup relativi ai moduli e alle schede verranno scritti in un file e memorizzati nella schedina di memoria insieme agli altri dati di backup.

Quando si preme l'interruttore di alimentazione, l'indicatore MCPWR lampeggia una volta, si accende durante l'operazione di scrittura e si spegne se la scrittura viene completata correttamente.

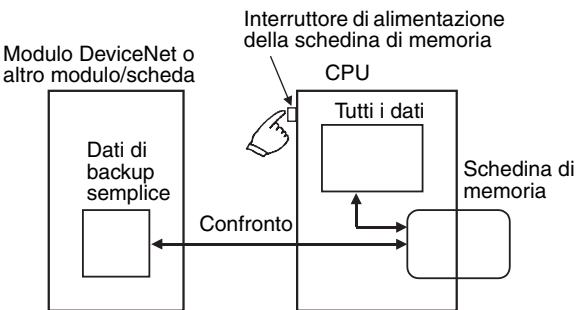
■ Ripristino dei dati

1,2,3...
1. Posizionare su ON il pin 7 del DIP switch della CPU.
2. Accendere il PLC. I file di backup verranno ripristinati su moduli e schede.

   I dati di backup sulla schedina di memoria relativi a moduli e schede specifici vengono ripristinati sui moduli e sulle schede corrispondenti.

Quando si preme l'interruttore di alimentazione, l'indicatore MCPWR lampeggia una volta, si accende durante l'operazione di lettura e si spegne se la lettura viene completata correttamente.

■ Confronto dei dati

1,2,3...
1. Posizionare su OFF il pin 7 del DIP switch della CPU.
2. Tenere premuto l'interruttore di alimentazione della schedina di memoria per tre secondi.

   I dati di backup della schedina di memoria verranno confrontati con i dati presenti nei moduli e nelle schede.

Quando si preme l'interruttore di alimentazione, l'indicatore MCPWR lampeggia una volta, si accende durante l'operazione di confronto e si spegne se il confronto viene completato correttamente e i dati corrispondono.

Nota Il tempo necessario per un'operazione di backup semplice con CS1W-SCU21-V1è maggiore rispetto a quello richiesto senza CS1W-SCU21-V1, come riportato nelle seguenti tabelle.

Tempo aggiuntivo quando il file delle impostazioni del Modulo CPU bus (BACKUP□□.PRM) nella schedina di memoria è 60 KB

Modalità operativa	Tempo aggiuntivo durante la scrittura in una schedina di memoria	Tempo aggiuntivo durante la verifica di una schedina di memoria	Tempo aggiuntivo durante la lettura da una schedina di memoria
PROGRAM	Circa 25 s	Circa 10 s	Circa 4 s
RUN	Circa 1 min e 30 s	Circa 30 s	Circa 4 s

Tempo aggiuntivo quando il file delle impostazioni del Modulo CPU bus (BACKUP□□.PRM) nella schedina di memoria è 128 KB

Modalità operativa	Tempo aggiuntivo durante la scrittura in una schedina di memoria	Tempo aggiuntivo durante la verifica di una schedina di memoria	Tempo aggiuntivo durante la lettura da una schedina di memoria
PROGRAM	Circa 40 s	Circa 14 s	Circa 8 s
RUN	Circa 2 min e 30 s	Circa 1 min	Circa 8 s

Nota

1. Verificare che i moduli e le schede funzionino correttamente prima di eseguire le operazioni descritte. Se schede e moduli non funzionano in modo corretto, le operazioni di scrittura, lettura e confronto non verranno eseguite.
2. Prima di eseguire un'operazione di backup semplice per moduli o schede specifici, accertarsi che la CPU sia in modalità PROGRAM o che l'esecuzione del backup semplice non abbia effetti negativi sulle istruzioni che utilizzano i numeri delle porte di comunicazione. Quando viene eseguito il backup dei dati da moduli o schede specifici, viene eseguita una ricerca per determinare la prima porta di comunicazione disponibile, a partire dalla porta 0. Se il numero della porta è identico a quello utilizzato da un'istruzione di comunicazione di rete, questa non verrà eseguita fino a quando l'operazione di backup semplice non sarà stata completata.

Precauzioni per la modifica dello stato di allocazione degli I/O durante operazioni di backup/ripristino

Lo stato di allocazione degli I/O dipende dalle versioni delle CPU di origine e di destinazione quando si utilizza una CPU della serie CJ per creare i file dei parametri di backup (BKUP.STD), per salvarli in una memory card e quindi per eseguire il backup o il ripristino in un'altra CPU della serie CJ. Le modifiche allo stato di allocazione degli I/O per le combinazioni di diverse versioni della CPU sono riportate nella seguente tabella.

CPU di origine		Stato di allocazione degli I/O di origine	CPU di destinazione		
			Versione della CPU alla destinazione di backup/ripristino		
			CPU precedenti alla versione 2.0	CPU versione 2.0	CPU versione 3.0 o successiva
Versione della CPU all'origine di backup	CPU precedenti alla versione 2.0	Allocazione automatica	(Uguale) Allocazione automatica		
		Definita dall'utente	(Uguale) Definita dall'utente		
	CPU versione 2.0	Allocazione automatica	(Uguale) Allocazione automatica		
		Definita dall'utente	(Uguale) Definita dall'utente		
	CPU versione 3.0 o successiva	Allocazione automatica	(Uguale) Allocazione automatica		
		Definita dall'utente	(Uguale) Definita dall'utente		

Precauzioni per incongruenze che si verificano durante il confronto di backup

Quando si utilizza una CPU della serie CJ versione 2.0 o precedente con una CPU della serie CJ versione 3.0, è possibile che si verifichino degli errori durante il confronto dei dati dei parametri con i dati di backup ripristinati dal file di backup semplice precedentemente creato.

<table>
<tr><th colspan="2" rowspan="3">CPU di origine</th><th rowspan="3">Stato di allocazione degli I/O precedente</th><th colspan="3">CPU di destinazione</th></tr>
<tr><th colspan="3">Versione della CPU alla destinazione di backup/ripristino</th></tr>
<tr><th>CPU precedenti alla versione 2.0</th><th>CPU versione 2.0</th><th>CPU versione 3.0 o successiva</th></tr>
<tr><td rowspan="6">Versione della CPU all'origine di backup</td><td rowspan="2">CPU precedenti alla versione 2.0</td><td>Allocazione automatica</td><td rowspan="2">Verifica possibile</td><td rowspan="4">Verifica possibile</td><td rowspan="2">Errore di verifica</td></tr>
<tr><td>Definita dall'utente</td></tr>
<tr><td rowspan="2">CPU versione 2.0</td><td>Allocazione automatica</td><td rowspan="2">Verifica possibile</td><td rowspan="2">Verifica possibile</td></tr>
<tr><td>Definita dall'utente</td></tr>
<tr><td rowspan="2">CPU versione 3.0 o successiva</td><td>Allocazione automatica</td><td>Errore di verifica</td><td>Errore di verifica</td><td rowspan="2">Verifica possibile</td></tr>
<tr><td>Definita dall'utente</td><td>Verifica possibile</td><td>Verifica possibile</td></tr>
</table>

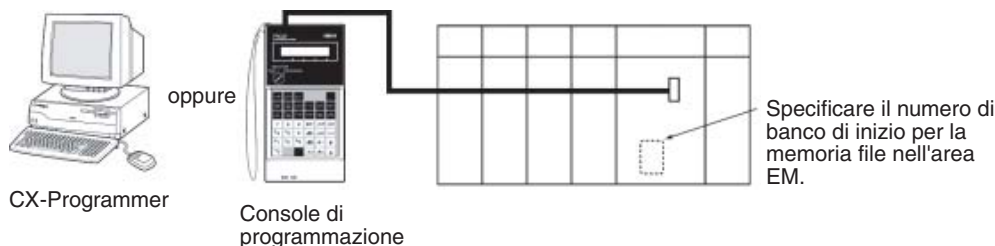
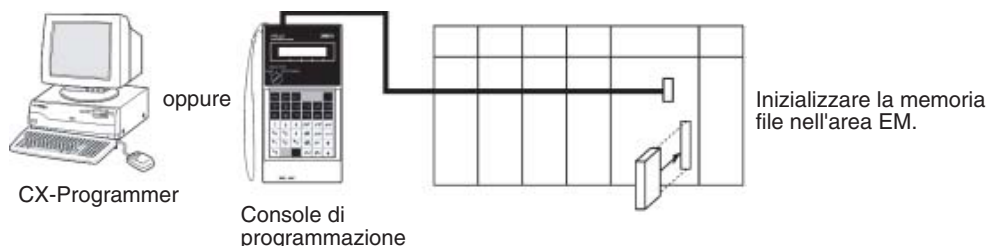
5-3 Utilizzo della memoria file

5-3-1 Inizializzazione dei supporti

schedine di memoria

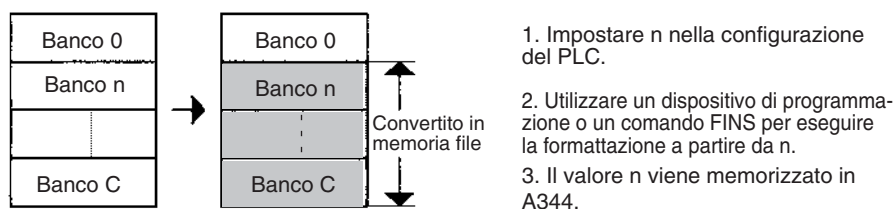
1,2,3... 1. Per inizializzare le schedine di memoria, utilizzare un dispositivo di programmazione, quale una Console di programmazione.

Memoria file nell'area EM

1,2,3... 1. Per abilitare la memoria file nell'area EM, utilizzare un dispositivo di programmazione, quale una Console di programmazione, e configurare le impostazioni necessarie nella configurazione del PLC, quindi impostare il numero di banco EM specifico per la memoria file nell'intervallo da 0 a C esadecimale.

2. Utilizzare un comando FINS o un dispositivo di programmazione diverso, non la Console di programmazione, per inizializzare la memoria file nell'area EM.

Inizializzazione di una singola memoria file nell'area EM

È possibile convertire un banco EM specifico in memoria file.

Nota Il numero massimo di banchi per le CPU della serie CJ è 6.

1. Impostare n nella configurazione del PLC.
2. Utilizzare un dispositivo di programmazione o un comando FINS per eseguire la formattazione a partire da n.
3. Il valore n viene memorizzato in A344.

È altresì possibile ripristinare le condizioni iniziali dei banchi EM utilizzati come memoria file.

1. Disabilitare la memoria file nella configurazione del PLC.
2. Se per la formattazione si utilizza un dispositivo di programmazione o un comando FINS, il contenuto della memoria a partire da n viene cancellato e impostato su 0000 esadecimale.
3. In A344 viene memorizzato il valore FFFF esadecimale per indicare che non è presente alcuna memoria file nell'area EM.

Nota: 1. I file eventualmente presenti vengono eliminati in questa fase.
2. Per le CPU della serie CJ è possibile specificare solo i banchi da 0 a 6.

Il numero del banco di inizio della memoria file può essere modificato.

1. Sostituire m a n nella configurazione del PLC.
2. Utilizzare un dispositivo di programmazione o un comando FINS per convertire in memoria file i banchi a partire da m.
Nota: il contenuto dei banchi da n a m-1 viene cancellato e impostato sul valore 0000 esadecimale.
3. Il valore m viene memorizzato in A344.

Nota: 1. I file eventualmente presenti vengono eliminati in questa fase.
2. Per le CPU della serie CJ è possibile specificare solo i banchi da 0 a 6.

Configurazione del PLC

Indirizzo	Nome	Descrizione	Impostazione iniziale
136	Banco di inizio della memoria file nell'area EM	0000 esadecimale: nessuno 0080 esadecimale: a partire dal banco 0 008C esadecimale: banco C L'area EM a partire dal numero di banco specificato verrà convertita in memoria file.	0000 esadecimale

Relè ausiliario speciale correlato

Nome	Indirizzo	Descrizione
Banco di inizio della memoria file nell'area EM	A344	Verrà memorizzato il numero di banco da cui inizia effettivamente l'area della memoria file nell'area EM. L'area EM dal numero di banco di inizio all'ultimo banco verrà convertita in memoria file. Il valore FFFF esadecimale indica che non è presente alcuna memoria file nell'area EM.

Lettura e scrittura di tabelle di simboli e commenti con CX-Programmer

Per trasferire tabelle di simboli o commenti creati in CX-Programmer da e verso la schedina di memoria o la memoria file nell'area EM, attenersi alla procedura descritta.

1,2,3...

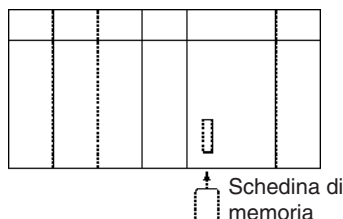
1. Inserire una schedina di memoria formattata nella CPU o formattare la memoria file nell'area EM.
2. Portare CX-Programmer in linea.
3. Scegliere ***Transfer*** (Trasferisci), quindi ***To PLC*** (Al PLC) o ***From PLC*** (Dal PLC) dal menu PLC.
4. Selezionare **Symbols** (Simboli) o **Comments** (Commenti) per i dati da trasferire.

5-3-2 Procedure operative

schedine di memoria

Utilizzo di un dispositivo di programmazione

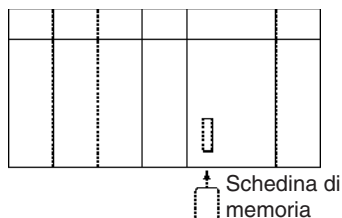
1,2,3...

1. Inserire una schedina di memoria nella CPU.

2. Inizializzare la schedina di memoria mediante un dispositivo di programmazione.

3. Utilizzare un dispositivo di programmazione per assegnare un nome ai dati della CPU (programma utente, memoria I/O, area dei parametri), quindi salvare i dati sulla schedina di memoria. Per trasferire il file dalla schedina di memoria alla CPU, utilizzare un dispositivo di programmazione.

Trasferimento automatico dei file all'avvio

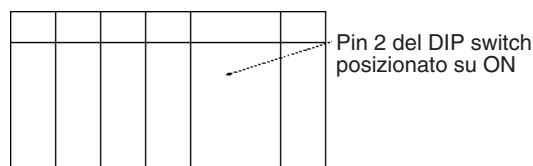
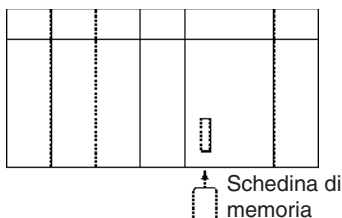
Procedura da utilizzare quando si trasferisce un file dell'area dei parametri

1,2,3...
1. Inserire una schedina di memoria inizializzata nella CPU.

2. Utilizzare un dispositivo di programmazione per scrivere sulla schedina di memoria i file per il trasferimento automatico all'avvio. Tali file comprendono il file del programma (AUTOEXEC.OBJ), il file dell'area dei parametri (AUTOEXEC.STD) e i file di dati relativi alla memoria I/O (AUTOEXEC.IOM, ATEXECDM.IOM e/o ATEXECE□.IOM).

Nota Sulla schedina di memoria devono essere presenti un file di programma utente e un file dell'area dei parametri.

3. Spegnere il PLC.
4. Impostare su ON il pin 2 del DIP switch (trasferimento automatico all'avvio).

Nota Se il pin 7 è impostato su ON e il pin 8 su OFF, la funzione di backup risulta abilitata e avrà la precedenza sulla funzione di trasferimento automatico all'avvio. Per eseguire il trasferimento automatico all'avvio, assicurarsi che i pin 7 e 8 siano posizionati su OFF.

5. Inserire la schedina di memoria nella CPU.

6. Accendere il PLC per leggere i file.

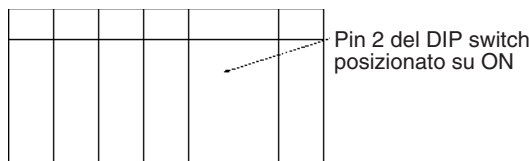
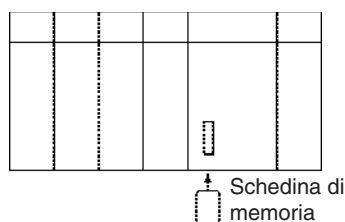
Procedura da utilizzare quando non si trasferisce un file dell'area dei parametri

1,2,3...
1. Inserire una schedina di memoria inizializzata nella CPU.
2. Utilizzare un dispositivo di programmazione per scrivere sulla schedina di memoria i file per il trasferimento automatico all'avvio, che comprendono il file del programma (REPLACE.OBJ) e i file di dati relativi alla memoria I/O (REPLACE.IOM, REPLCDM.IOM e/o REPLCDE□.IOM).

Nota non verrà trasferito alcun file dell'area dei parametri, anche se presente nella schedina di memoria.

3. Spegnere il PLC.
4. Impostare su ON il pin 2 del DIP switch (trasferimento automatico all'avvio).

Nota Se il pin 7 è impostato su ON e il pin 8 su OFF, la funzione di backup risulta abilitata e avrà la precedenza sulla funzione di trasferimento automatico all'avvio. Per eseguire il trasferimento automatico all'avvio, assicurarsi che i pin 7 e 8 siano posizionati su OFF.

5. Inserire la schedina di memoria nella CPU.

6. Accendere il PLC per leggere i file.

Utilizzo di FREAD(700), FWRIT(701) e CMND(490)

1,2,3...
1. Inserire una schedina di memoria nella CPU (già inizializzata).
2. Utilizzare l'istruzione FWRIT(701) per assegnare un nome al file nell'area specificata della memoria I/O, quindi salvare il file sulla schedina di memoria.

   Nota È possibile installare una schedina di memoria contenente file di dati in formato TXT o CSV nello slot per schede PLC di un personal computer utilizzando un adattatore per schedine di memoria HMC-AP001. In questo modo, sarà possibile elaborare i file di dati mediante un foglio elettronico utilizzando le funzioni standard di Windows (opzione non supportata dalle CPU CS1 della serie CS precedenti a EV1).
3. Utilizzare l'istruzione FREAD(700) per leggere i file dalla schedina di memoria alla memoria I/O della CPU.

È possibile eseguire operazioni sui file della schedina di memoria inviando comandi FINS alla CPU locale tramite l'istruzione CMND(490) (funzione non supportata dalle CPU CS1 serie CS precedenti a EV1).

Sostituzione del programma durante il funzionamento

1,2,3...
1. Inserire una schedina di memoria nella CPU (già inizializzata).
2. Scrivere la password del programma (A5A5 esadecimale) in A651 e il nome del file di programma nei canali da A654 ad A657.
3. Attivare il bit di inizio sostituzione (A65015).

Funzione di backup semplice

Sono disponibili tre operazioni di backup, ovvero backup dei dati sulla schedina di memoria, ripristino dei dati dalla schedina di memoria e confronto dei dati memorizzati nella schedina di memoria.

Backup dei dati dalla CPU alla schedina di memoria

1,2,3...
1. Inserire una schedina di memoria nella CPU (già inizializzata).
2. Posizionare il pin 7 del DIP switch della CPU su ON e il pin 8 su OFF.
3. Tenere premuto l'interruttore di alimentazione della schedina di memoria per tre secondi.
4. Verificare che l'indicatore MCPWR lampeggi una volta e quindi si spenga. Un comportamento diverso indica la presenza di errori durante il backup dei dati.

Ripristino dei dati dalla schedina di memoria alla CPU

1,2,3...
1. Inserire la schedina di memoria contenente i file di backup nella CPU.
2. Posizionare il pin 7 del DIP switch della CPU su ON e il pin 8 su OFF.
3. I file di backup verranno ripristinati all'accensione del PLC.
4. Verificare che l'indicatore MCPWR lampeggi una volta e quindi si spenga. Un comportamento diverso indica la presenza di errori durante il ripristino dei dati.

Confronto tra i dati della schedina di memoria e della CPU

1,2,3...
1. Inserire la schedina di memoria contenente i file di backup nella CPU.
2. Posizinare su OFF i pin 7 e 8 del DIP switch della CPU.
3. Tenere premuto l'interruttore di alimentazione della schedina di memoria per tre secondi.
4. Se l'indicatore MCPWR lampeggia una volta e quindi si spegne, è segno che i dati corrispondono.

Nota In caso di errore durante la scrittura o il confronto dei dati, l'indicatore MCPWR continua a lampeggiare. Se si preme l'interruttore di alimentazione della schedina di memoria, l'indicatore MCPWR smetterà di lampeggiare e rimarrà acceso.

Nella tabella riportata di seguito viene indicato il tempo richiesto per le operazioni di backup con un programma da 20.000 step e un tempo di ciclo di 10 ms in modalità RUN.

Modalità	Backup	Ripristino	Confronto
PROGRAM	Circa 50 s	Circa 30 s	Circa 7 s
RUN	Circa 5 min	Circa 2 min	Circa 7 s

Nella tabella riportata di seguito viene indicato il tempo richiesto per le operazioni di backup con un programma da 30.000 step e un tempo di ciclo di 10 ms in modalità RUN.

Modalità	Backup	Ripristino	Confronto
PROGRAM	Circa 50 s	Circa 30 s	Circa 7 s
RUN	Circa 5 min e 30 s	Circa 2 min e 40 s	Circa 7 s

Nella tabella riportata di seguito viene indicato il tempo richiesto per le operazioni di backup con un programma da 250.000 step e un tempo di ciclo di 12 ms in modalità RUN.

Modalità	Backup	Ripristino	Confronto
PROGRAM	Circa 1 min e 30 s	Circa 1 min e 30 s	Circa 20 s
RUN	Circa 13 min	Circa 7 min e 30 s	Circa 20 s

Creazione di file di commento e di tabelle di variabili

Per creare file di tabelle di variabili o file di commento sulle schedine di memoria o nella memoria file nell'area EM, utilizzare CX-Programmer e attenersi alla procedura descritta.

1,2,3...

1. Inserire una schedina di memoria formattata nella CPU o formattare la memoria file nell'area EM.
2. Portare CX-Programmer in linea.
3. Scegliere ***Transfer*** (Trasferisci), quindi ***To PLC*** (Al PLC) o ***From PLC*** (Dal PLC) dal menu PLC.
4. Selezionare **Symbols** (Simboli) o **Comments** (Commenti) per i dati da trasferire.

Nota Se nella CPU è installata una schedina di memoria, è possibile trasferire i dati solo con la schedina di memoria (non è possibile utilizzare la memoria file nell'area EM).

Memoria file nell'area EM

Utilizzo di un dispositivo di programmazione

1,2,3...

1. Nella configurazione del PLC, specificare il banco EM di inizio da convertire in memoria file.
2. Utilizzare un dispositivo di programmazione per inizializzare la memoria file nell'area EM.
3. Utilizzare un dispositivo di programmazione per assegnare un nome ai dati della CPU (programma utente, memoria I/O, area dei parametri), quindi salvare i dati nella memoria file nell'area EM.
4. Utilizzare un dispositivo di programmazione per trasferire i file dalla memoria file nell'area EM alla CPU.

Utilizzo di FREAD(700), FWRIT(701) e CMND(490)

1,2,3...

1. Nella configurazione del PLC, specificare il banco EM di inizio da convertire in memoria file.
2. Utilizzare un dispositivo di programmazione per inizializzare la memoria file nell'area EM.
3. Utilizzare l'istruzione FWRIT(701) per assegnare un nome al file nell'area specificata della memoria I/O, quindi salvare il file nella memoria file nell'area EM.
4. Utilizzare l'istruzione FREAD(700) per leggere il file dalla memoria file nell'area EM alla memoria I/O della CPU.

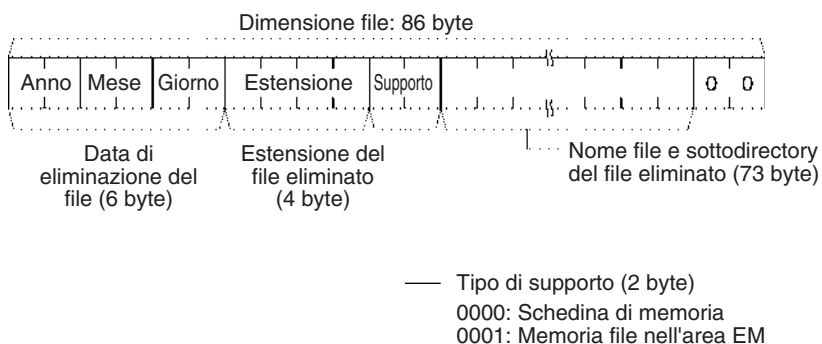
È possibile eseguire operazioni sulla memoria file nell'area EM inviando comandi FINS alla CPU locale tramite l'istruzione CMND(490).

5-3-3 Cadute di tensione durante l'accesso alla memoria file

Se si verifica una caduta di tensione mentre la CPU accede alla memoria file (schedina di memoria o memoria file nell'area EM), il contenuto della schedina di memoria potrebbe risultare non corretto. È possibile che un file in corso di aggiornamento non venga sovrascritto in modo corretto e, in alcuni casi, che la schedina di memoria stessa venga danneggiata.

Il file interessato verrà automaticamente eliminato dal sistema all'avvio successivo. e verrà attivato il flag di notifica di eliminazione file corrispondente, ovvero A39507 per la schedina di memoria e A39506 per la memoria file nell'area EM. Il flag verrà disattivato allo spegnimento del sistema.

Quando un file viene eliminato, nella directory principale della schedina di memoria o della memoria file nell'area EM viene creato un file di log di eliminazione (DEL_FILE.IOM). È possibile leggere tale file di log con CX-Programmer o l'istruzione FREAD(700) per ottenere le seguenti informazioni: data di eliminazione del file, tipo di memoria file (supporto) utilizzato, sottodirectory, nome del file ed estensione. Se necessario, ricreare o ricopiare il file eliminato.

Lo schema riportato di seguito illustra la struttura del file di log di eliminazione.

CAPITOLO 6
Funzioni avanzate

In questo capitolo vengono descritte in dettaglio le seguenti funzioni avanzate: tempo di ciclo ed elaborazione veloce, funzioni del registro indice, funzioni per le comunicazioni seriali, funzioni di avvio e manutenzione, funzioni di diagnostica e debug, funzioni dei dispositivi di programmazione e le impostazioni dei tempi di risposta di ingresso per i Moduli di I/O di base.

6-1 Tempo di ciclo ed elaborazione veloce

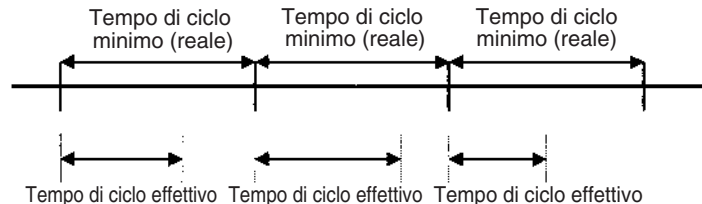
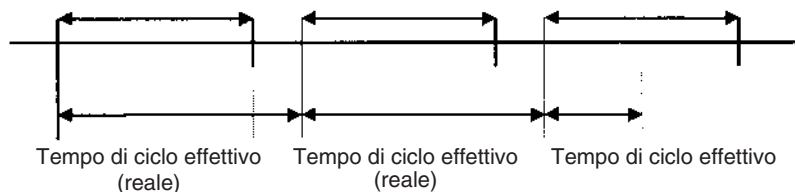
In questa sezione vengono descritte le seguenti funzioni:

- Funzione del tempo di ciclo minimo
- Funzione del tempo di ciclo massimo (tempo di ciclo limite)
- Monitoraggio del tempo di ciclo
- Ingressi a risposta rapida
- Funzioni di interrupt
- Metodi di aggiornamento degli I/O
- Disattivazione dell'aggiornamento ciclico di un Modulo di I/O speciale
- Miglioramento della risposta di aggiornamento per data link e altri dati relativi ai Moduli CPU bus (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)
- Riduzione della variazione del tempo di ciclo tramite l'esecuzione in background di operazioni sui dati (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)

6-1-1 Tempo di ciclo minimo

È possibile impostare un tempo di ciclo minimo (o fisso) per i PLC della serie CS/CJ (vedere nota). Le variazioni dei tempi di risposta degli I/O possono quindi essere eliminate ripetendo il programma con un tempo di ciclo fisso.

Nota è anche possibile fissare il tempo di ciclo per le CPU CS1D per sistemi a singola CPU o per le CPU CS1-H, CJ1-H o CJ1M utilizzando una modalità di elaborazione parallela.

Il tempo di ciclo minimo (da 1 a 32.000 ms) viene specificato in unità da 1 ms nella configurazione del PLC.

Se il tempo di ciclo effettivo è superiore al tempo di ciclo minimo, la funzione del tempo di ciclo minimo non verrà applicata e il tempo di ciclo varierà da ciclo a ciclo.

Configurazione del PLC

Indirizzo	Nome	Impostazione	Valore predefinito
208 Bit: da 0 a 15	Tempo di ciclo minimo	Da 0001 a 7D00: da 1 a 32.000 ms (unità: 1 ms)	0000 (nessun minimo)

6-1-2 Tempo di ciclo massimo (tempo di ciclo limite)

Se il tempo di ciclo (vedere nota) supera il tempo di ciclo massimo impostato, verrà attivato il flag di superamento del tempo di ciclo massimo (A40108) e il funzionamento del PLC verrà interrotto.

Nota In questo caso, il tempo di ciclo corrisponde al tempo di esecuzione del programma quando si utilizza una modalità di elaborazione parallela per le CPU CS1-H, CJ1-H o CJ1M oppure per le CPU CS1D per sistemi a singola CPU.

Configurazione del PLC

Indirizzo	Nome	Impostazione	Valore predefinito
209 Bit: 15	Attivazione del tempo di ciclo limite	0: predefinito (1 s) 1: Bit da 0 a 14	0001 (1 s)
209 Bit: da 0 a 14	Impostazione del tempo di ciclo limite (attivato quando il bit 15 è impostato su 1)	Da 001 a FA0: da 1 a 40.000 ms (unità: 10 ms)	

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Descrizione
Flag di superamento del tempo di ciclo massimo	A40108	Se il tempo di ciclo supera il tempo di ciclo limite impostato, viene attivato il flag A40108 e il funzionamento della CPU viene interrotto. Il "tempo di ciclo" corrisponde al tempo di esecuzione del programma quando si utilizza una modalità di elaborazione parallela per le CPU CS1-H, CJ1-H o CJ1M oppure per le CPU CS1D per sistemi a singola CPU.

Nota Se il ciclo di gestione delle periferiche supera 2,0 s per le CPU CS1-H, CJ1-H o CJ1M oppure per le CPU CS1D per sistemi a singola CPU in modalità di elaborazione parallela, si verificherà un errore di superamento del tempo di ciclo di gestione delle periferiche e il funzionamento della CPU verrà interrotto. In tal caso, viene attivato il flag A40515 (flag di superamento del tempo del ciclo di gestione delle periferiche).

6-1-3 Monitoraggio del tempo di ciclo

Il tempo di ciclo massimo e il tempo di ciclo attuale vengono memorizzati nell'area ausiliaria a ogni ciclo. Per le CPU CS1-H, CJ1-H o CS1D per sistemi a singola CPU in modalità di elaborazione parallela, vengono memorizzati anche i tempi di esecuzione del programma.

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Descrizione
Tempo di ciclo massimo (tempo di esecuzione del programma per CPU CS1-H, CJ1-H o CJ1M in modalità di elaborazione parallela)	A262 e A263	Viene memorizzato ad ogni ciclo come valore binario a 32 bit nel seguente intervallo: da 0 a 429.496.729,5 ms, in unità di 0,1 ms (da 0 a FFFF FFFF)
Tempo di ciclo attuale (tempo di esecuzione del programma per CPU CS1-H, CJ1-H o CJ1M in modalità di elaborazione parallela)	A264 e A265	Viene memorizzato ad ogni ciclo come valore binario a 32 bit nel seguente intervallo: da 0 a 429.496.729,5 ms, in unità di 0,1 ms (da 0 a FFFF FFFF)

È possibile utilizzare un dispositivo di programmazione (CX-Programmer o Console di programmazione) per leggere la media dei tempi di ciclo degli ultimi 8 cicli.

Riduzione del tempo di ciclo

I metodi elencati di seguito consentono di ridurre il tempo di ciclo nei PLC della serie CS/CJ.

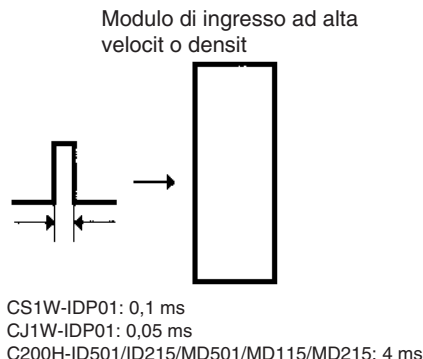
1,2,3...
1. Mettere in attesa i task che non sono in esecuzione.
2. Saltare le sezioni del programma che non vengono eseguite utilizzando le istruzioni JMP(004) e JME(005).

Per le CPU CS1-H o CJ1-H o per le CPU CS1D per sistemi a singola CPU in modalità di elaborazione parallela, il tempo del ciclo di gestione delle periferiche viene memorizzato in A268 a ogni ciclo di gestione.

6-1-4 Ingressi ad alta velocità

Per ricevere impulsi più brevi del tempo di ciclo, utilizzare il Modulo di ingresso ad alta velocità CS1W-IDP01 o gli ingressi ad alta velocità dei Moduli di I/O ad alta densità C200H-ID501/ID215 e C200H-MD501/MD115/MD215.

Gli ingressi ad alta velocità consentono di ricevere impulsi con un'ampiezza di impulso (tempo di attivazione) di 1 ms o 4 ms per i Moduli di ingresso ad alta densità C200H e di 0,1 ms per il Modulo di ingresso ad alta velocità CS1W-IDP01.

CS1W-IDP01: 0,1 ms
CJ1W-IDP01: 0,05 ms
C200H-ID501/ID215/MD501/MD115/MD215: 4 ms

6-1-5 Funzioni di interrupt

È possibile eseguire task ad interrupt per le condizioni riportate di seguito. Per ulteriori informazioni, fare riferimento al capitolo *4-3 Task ad interrupt.*

Nota Le CPU CS1D per sistemi a due CPU non supportano gli interrupt. Con le CPU CS1D è possibile utilizzare i task ad interrupt solo come task ciclici supplementari. In pratica, non è possibile utilizzare alcun altro tipo di task ad interrupt.

Interrupt di I/O (task ad interrupt da 100 a 131)

Un task ad interrupt di I/O viene eseguito quando un Modulo di interrupt di ingresso invia il segnale di ingresso corrispondente (sul fronte di salita del segnale o, per i Moduli di interrupt di ingresso della serie CS/CJ, sia sul fronte di salita che di discesa).

Interrupt programmati (task ad interrupt 2 e 3)

Un task ad interrupt programmato viene eseguito a intervalli regolari.

Interrupt di spegnimento (task ad interrupt 1)

Questo task viene eseguito quando viene interrotta l'alimentazione.

Interrupt esterni (task ad interrupt da 0 a 255)

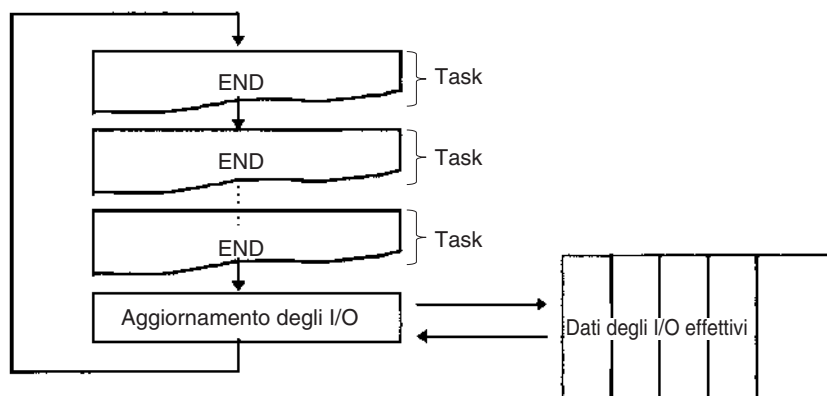
Un task ad interrupt esterno viene eseguito alla ricezione di un interrupt inviato da un Modulo di I/O speciale, un Modulo CPU bus o una scheda interna.

Nota Per attivare i task ad interrupt, è possibile utilizzare gli ingressi di interrupt integrati e gli ingressi del contatore ad alta velocità di una CPU CJ1M. Per ulteriori informazioni, fare riferimento al *Manuale dell'operatore degli I/O integrati della serie CJ*.

6-1-6 Metodi di aggiornamento degli I/O

Per le CPU della serie CS/CJ sono disponibili tre metodi di aggiornamento dei dati con i Moduli di I/O di base e Moduli di I/O speciale: l'aggiornamento ciclico, l'aggiornamento immediato e l'esecuzione dell'istruzione IORF(097).

1. Aggiornamento ciclico

L'aggiornamento degli I/O ha luogo dopo l'esecuzione di tutte le istruzioni dei task eseguibili. A tale proposito, è possibile impostare la configurazione del PLC in modo da disattivare l'aggiornamento ciclico di singoli Moduli di I/O speciale.

2. Aggiornamento immediato

Quando si specifica un indirizzo dell'area degli I/O come operando per variazioni di aggiornamento immediato di un'istruzione, i dati di tale operando vengono aggiornati all'esecuzione dell'istruzione. Le istruzioni di aggiornamento immediato consentono di aggiornare i dati allocati ai Moduli di I/O di base.
È possibile eseguire l'aggiornamento immediato anche per gli I/O integrati delle CPU CJ1M.

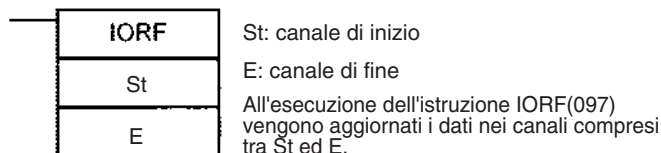
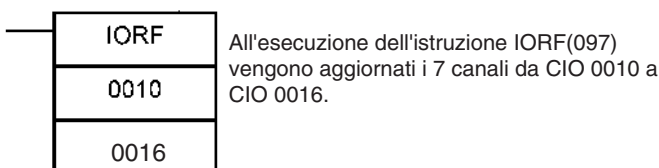
Nota

1. Quando l'istruzione contiene un bit operando, viene aggiornato l'intero canale che contiene tale bit. Quando l'istruzione contiene un canale operando, tale canale viene aggiornato.
2. I dati di origine e di ingresso vengono aggiornati subito prima dell'esecuzione dell'istruzione. I dati di destinazione e di uscita vengono aggiornati subito dopo l'esecuzione dell'istruzione.
3. Poiché i tempi di esecuzione per le variazioni di aggiornamento immediato risultano più lunghi rispetto alle normali variazioni di istruzione, il tempo di ciclo risulterà più lungo. Per ulteriori informazioni, fare riferimento al capitolo *10-5 Tempi di esecuzione delle istruzioni e numero di step* nel *Manuale dell'operatore*.

4. Le CPU CS1D per sistemi a due CPU non supportano l'aggiornamento immediato.

3. Esecuzione delle istruzioni IORF(097) e DLNK(226)

■ **Istruzione IORF(097): I/O REFRESH (aggiornamento degli I/O)**

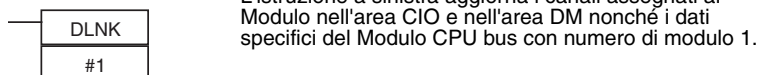
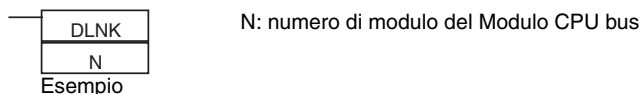
È possibile utilizzare l'istruzione IORF(097) per aggiornare un intervallo di canali di I/O all'esecuzione dell'istruzione. IORF(097) consente di aggiornare i dati allocati a Moduli di I/O di base e Moduli di I/O speciale.

L'esempio fornito di seguito illustra l'utilizzo di IORF(097) per l'aggiornamento di 8 canali di dati di I/O.

Quando è necessaria una risposta ad alta velocità per l'ingresso e l'uscita da un calcolo, utilizzare IORF(097) subito prima e subito dopo l'istruzione di calcolo.

Nota L'istruzione IORF(097) presenta un tempo di esecuzione relativamente lungo e tale tempo aumenta in modo proporzionale al numero di canali aggiornati. Pertanto, anche il tempo di ciclo aumenta in modo significativo. Per ulteriori informazioni, fare riferimento al capitolo *10-5 Tempi di esecuzione delle istruzioni e numero di step* nel *Manuale dell'operatore*.

■ **Istruzione DLNK(226): aggiornamento degli I/O dei Moduli CPU bus (solo CPU CS1-H, CJ1-H, CJ1M o CS1D)**

L'istruzione DLNK(226) viene utilizzata per aggiornare i dati per un Modulo CPU bus, specificando il numero di modulo. Vengono aggiornati i seguenti dati:

- Canali assegnati al Modulo nell'area CIO
- Canali assegnati al Modulo nell'area DM
- Dati specifici del Modulo (vedere nota)

Nota I dati specifici del Modulo CPU bus comprendono i data link per i Moduli Controller Link o SYSMAC LINK, oltre agli I/O remoti per i Moduli DeviceNet.

Esempio di applicazione: con un tempo di ciclo lungo, l'intervallo di aggiornamento per i data link Controller Link può risultare molto elevato. È possibile ridurre tale intervallo eseguendo DLNK(226) per il Modulo Controller Link in modo da aumentare la frequenza di aggiornamento dei data link.

6-1-7 Disattivazione dell'aggiornamento ciclico di un Modulo di I/O speciale

Dieci canali nell'area relativa ai Moduli di I/O speciale (da CIO 2000 a CIO 2959) vengono assegnati a ciascun Modulo di I/O speciale in base al numero di modulo impostato sul pannello frontale. I dati vengono aggiornati tra quest'area e la CPU a ogni ciclo durante l'aggiornamento degli I/O, ma è possibile disattivare l'aggiornamento ciclico per singoli Moduli nella configurazione del PLC.

In genere l'aggiornamento ciclico viene disattivato per tre motivi.

1,2,3...

1. È possibile disattivare l'aggiornamento ciclico per i Moduli di I/O speciale quando il tempo di ciclo risulta troppo lungo a causa dell'elevato numero di Moduli di I/O speciale installati.
2. Se il tempo di aggiornamento di I/O è troppo breve, è possibile che l'elaborazione interna del Modulo non riesca a stare al passo e che, di conseguenza, venga attivato il flag di errore del modulo di I/O speciale (A40206). In questo scenario è possibile che il Modulo di I/O speciale non funzioni in modo corretto.
   Per risolvere il problema, è possibile estendere il tempo di ciclo impostando un tempo di ciclo minimo nella configurazione del PLC oppure disattivare l'aggiornamento di I/O ciclico per il Modulo di I/O speciale.
3. Si consiglia di disattivare sempre l'aggiornamento ciclico qualora l'aggiornamento del Modulo di I/O speciale venga eseguito nell'ambito di un task ad interrupt mediante l'istruzione IORF(097). Se per il medesimo Modulo vengono eseguiti contemporaneamente l'aggiornamento ciclico e l'aggiornamento IORF(097), si verificherà un errore del task ad interrupt e verrà attivato il flag di errore del task ad interrupt (A40213).

Quando si disattiva l'aggiornamento ciclico, è possibile aggiornare i dati del Modulo di I/O speciale durante l'esecuzione del programma utilizzando l'istruzione IORF(097).

Configurazione del PLC

I bit di disattivazione dell'aggiornamento ciclico per i Moduli di I/O speciale da 0 a 95 corrispondono direttamente ai 96 bit degli indirizzi da 226 a 231.

Indirizzo	Nome	Impostazione	Valore predefinito
226 bit 0	Bit di disattivazione dell'aggiornamento ciclico per il Modulo di I/O speciale 0	0: abilitato 1: disabilitato	0 (abilitato)
:	:	:	:
231 bit 15	Bit di disattivazione dell'aggiornamento ciclico per il Modulo di I/O speciale 95	0: abilitato 1: disabilitato	0 (abilitato)

6-1-8 Miglioramento della risposta di aggiornamento per i dati dei Moduli CPU bus

Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

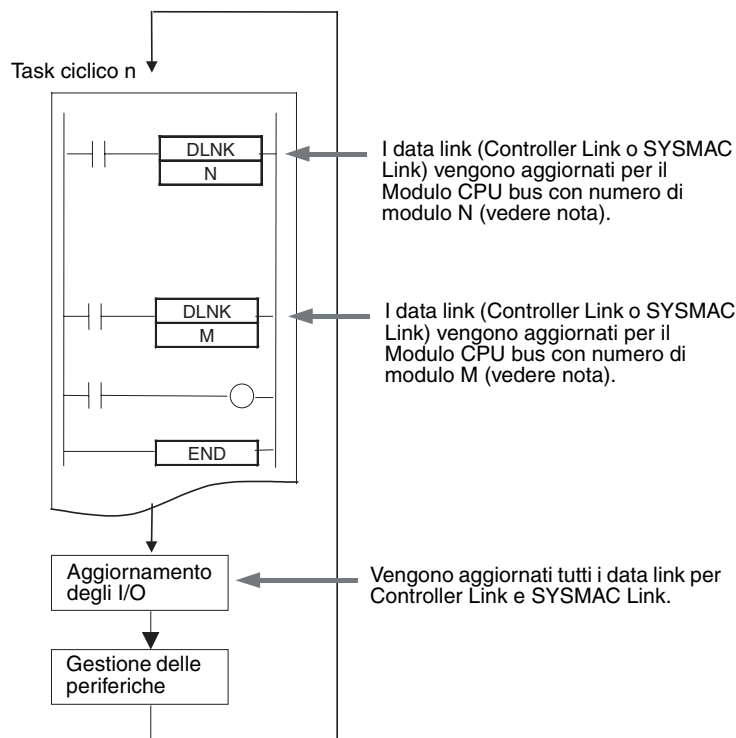
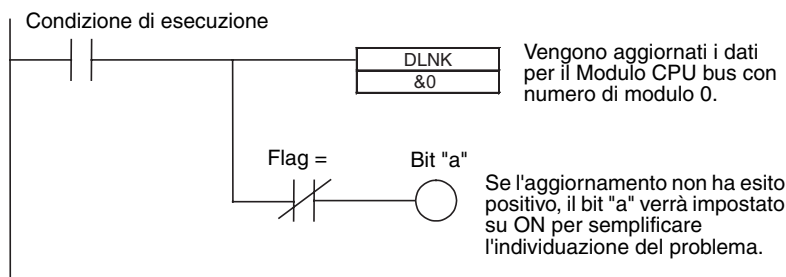
In genere i data link e altri dati speciali per i Moduli CPU bus vengono aggiornati insieme ai canali delle aree CIO e DM assegnati ai Moduli durante il periodo di aggiornamento degli I/O successivo all'esecuzione del programma.

Nella seguente tabella sono riportati alcuni esempi di dati speciali per i Moduli CPU bus.

Moduli	Dati speciali
Moduli Controller Link e SYSMAC LINK	Data link di Controller Link e SYSMAC LINK (inclusi i link impostati automaticamente e definiti dall'utente)
Moduli DeviceNet della serie CS/CJ	Comunicazione di I/O remoti DeviceNet (incluse le assegnazioni fisse e definite dall'utente)

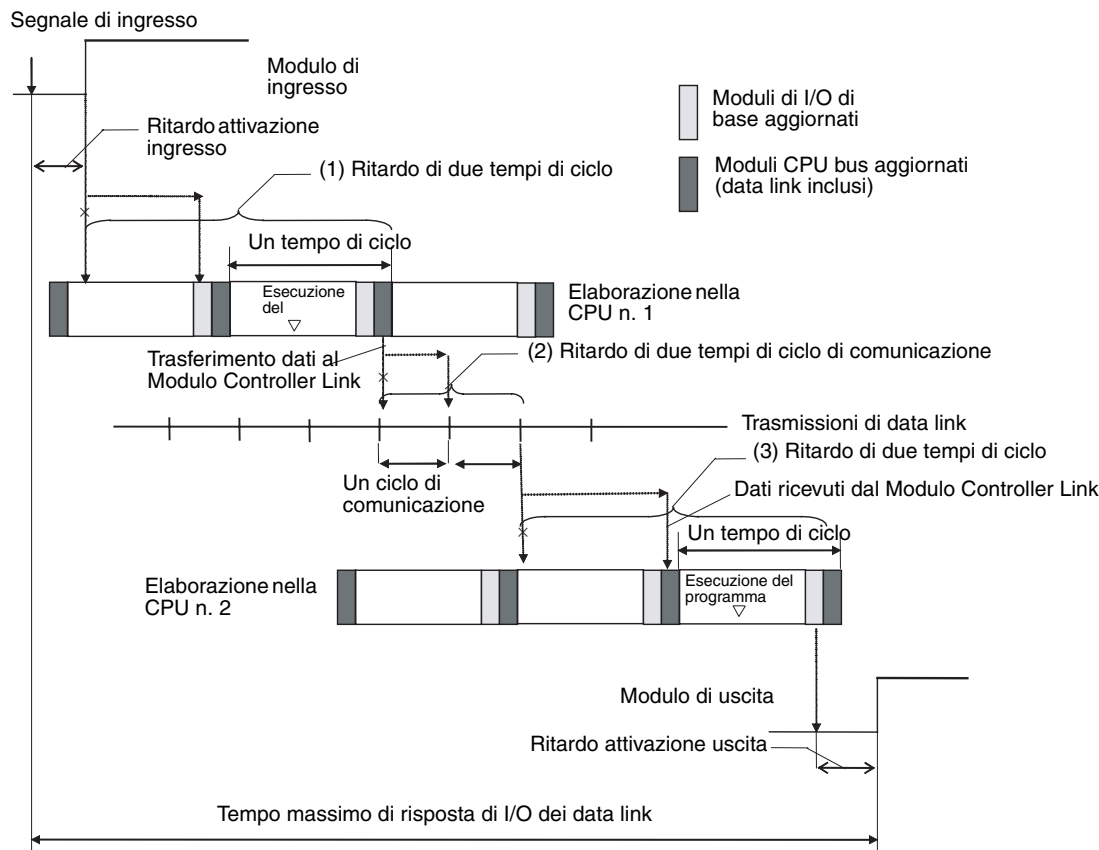
Per migliorare la risposta di aggiornamento dei dati speciali relativi a Moduli CPU bus per le CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile utilizzare le funzioni indicate di seguito.

- Riduzione del tempo di ciclo tramite l'impiego della modalità di elaborazione parallela o di istruzioni ad alta velocità (la modalità di elaborazione parallela non è supportata dalle CPU CS1D per sistemi a due CPU).
- Esecuzione dell'istruzione DLNK(226) per l'aggiornamento di Moduli CPU bus specifici, indicando i relativi numeri di modulo. DLNK(226) può essere utilizzata più di una volta nella compilazione del programma.

Nota 1. Tempi di ciclo più lunghi (ad esempio, 100 ms) comportano l'aumento dell'intervallo tra gli aggiornamenti dei data link. In tal caso, è possibile utilizzare l'istruzione DLNK(226), come illustrato dall'esempio riportato di seguito.

Nota Se viene eseguita DLNK(226) per un Modulo CPU bus su cui è in corso un aggiornamento dei dati, i dati non verranno aggiornati e il flag di uguaglianza verrà disattivato. Di norma, il flag di uguaglianza viene programmato come illustrato di seguito per verificare la corretta esecuzione dell'aggiornamento.

2. L'istruzione IORF(097) viene utilizzata per aggiornare i dati per Moduli di I/O di base e Moduli di I/O speciale. L'istruzione DLNK(226) viene utilizzata per l'aggiornamento di Moduli CPU bus (canali delle aree CIO e DM assegnati ai Moduli e dati speciali dei Moduli).

6-1-9 Tempo massimo di risposta di I/O dei data link

Elaborazione normale

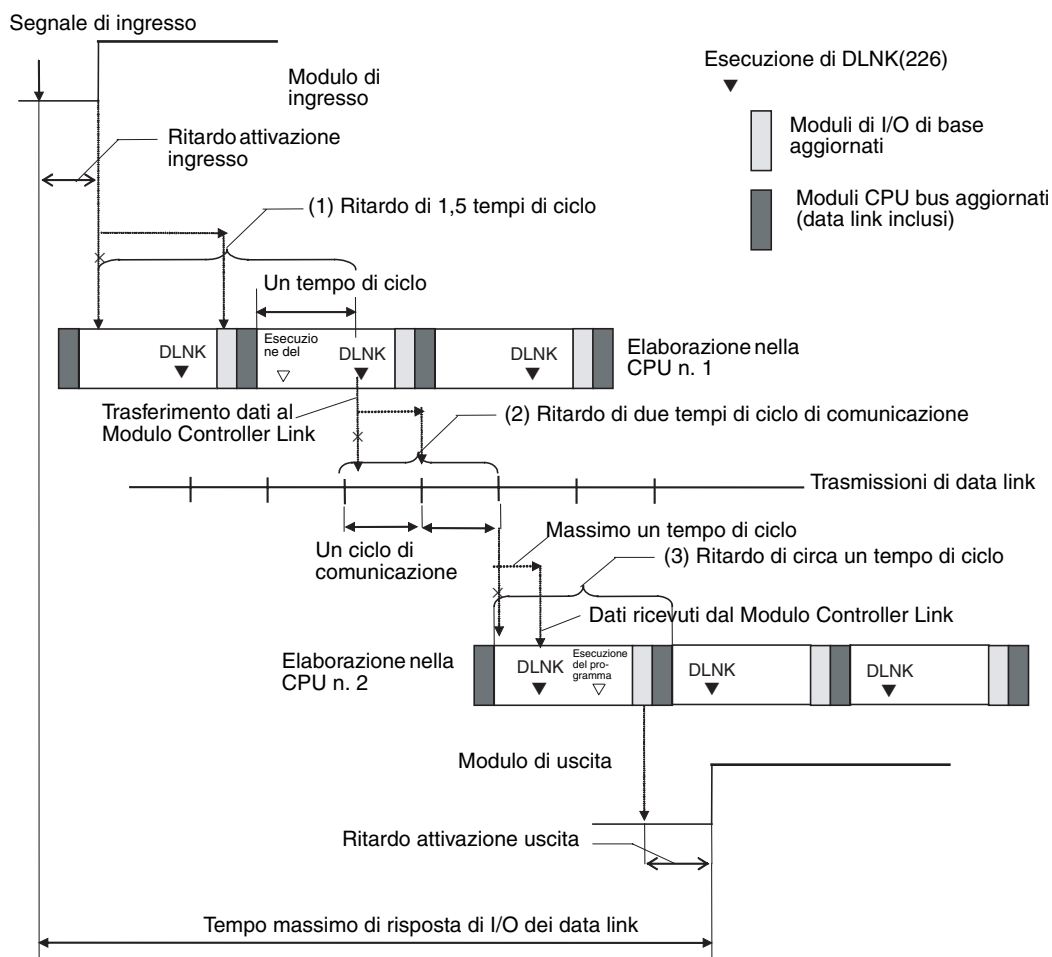
Lo schema riportato di seguito illustra il flusso dei dati che produrrà il tempo massimo di risposta di I/O dei data link quando non si utilizza l'istruzione DLNK(226).

In tre dei punti illustrati nello schema l'elaborazione viene ritardata, aumentando il tempo di risposta di I/O dei data link.

1,2,3...

1. Il segnale di ingresso giunge al PLC (CPU n. 1) dopo l'aggiornamento degli I/O, determinando un ritardo pari a un ciclo prima della lettura del segnale nel PLC. I Moduli CPU bus vengono aggiornati dopo l'esecuzione del programma, determinando un ritardo totale di due tempi di ciclo.
2. Lo scambio dei dati ha luogo subito dopo il passaggio del token da parte del PLC (nodo di polling), determinando un ritardo massimo di un tempo di ciclo di comunicazione prima del trasferimento dei dati durante l'elaborazione dei data link. Dopo la ricezione del token, si verifica un ulteriore ritardo pari a un tempo di ciclo di comunicazione al massimo, portando così il ritardo totale a un massimo di due tempi di ciclo di comunicazione.
3. I dati trasferiti durante l'elaborazione dei data link giungono al PLC (CPU n. 2) dopo lo scambio di dati. Pertanto, i dati non verranno letti nel PLC fino al successivo scambio, determinando un ritardo massimo di un ciclo. I Moduli CPU bus vengono aggiornati dopo l'esecuzione del programma, determinando un ritardo totale di due tempi di ciclo.

L'equazione per il calcolo del tempo massimo di risposta di I/O dei data link viene espressa come segue:

Ritardo attivazione ingresso	1,5 ms
Tempo di ciclo del PLC alla CPU n. 1 × 2	25 ms × 2
Tempo di ciclo di comunicazione × 2	10 ms × 2
Tempo di ciclo del PLC alla CPU n. 2 × 2	20 ms × 2
Ritardo attivazione uscita	15 ms
Totale (tempo di risposta di I/O dei data link)	126,5 ms

Utilizzo di DLNK(226) Lo schema riportato di seguito illustra il flusso dei dati che produrrà il tempo massimo di risposta di I/O dei data link quando si utilizza l'istruzione DLNK(226).

In tre dei punti illustrati nello schema l'elaborazione viene ritardata, aumentando il tempo di risposta di I/O dei data link.

Nota Nell'esempio si presume che l'istruzione DNLK(226) sia stata inserita dopo altre istruzioni nel programma di entrambe le CPU.

1,2,3...

1. Il segnale di ingresso giunge al PLC (CPU n. 1) dopo l'aggiornamento degli I/O, determinando un ritardo pari a un ciclo prima della lettura del segnale nel PLC. I Moduli CPU bus vengono aggiornati durante l'esecuzione del programma, riducendo il ritardo totale di circa 1,5 tempi di ciclo.
2. Lo scambio dei dati ha luogo subito dopo il passaggio del token da parte del PLC (nodo di polling), determinando un ritardo massimo di un tempo di ciclo di comunicazione prima del trasferimento dei dati durante l'elaborazione dei data link. Dopo la ricezione del token, si verifica un ulteriore ritardo pari a un tempo di ciclo di comunicazione al massimo, portando così il ritardo totale a un massimo di due tempi di ciclo di comunicazione.
3. Anche se i dati trasferiti durante l'elaborazione dei data link giungono al PLC (CPU n. 2) dopo l'aggiornamento degli I/O, l'istruzione DLNK(226) aggiorna i dati in modo che vengano letti nel PLC senza determinare alcun ritardo, che potrebbe essere pari a un ciclo al massimo. I Moduli di I/O di base vengono aggiornati dopo l'esecuzione del programma, determinando un ritardo totale di circa un tempo di ciclo.

L'equazione per il calcolo del tempo massimo di risposta di I/O dei data link viene espressa come segue:

Ritardo attivazione ingresso	1,5 ms	---
Tempo di ciclo del PLC alla CPU n. 1 × 1,5	25 ms × 1,5	Più veloce di 12,5 ms (25 ms x 0,5)
Tempo di ciclo di comunicazione × 2	10 ms × 2	---
Tempo di ciclo del PLC alla CPU n. 2 × 1	20 ms × 1	Più veloce di 20 ms (20 ms x 1)
Ritardo attivazione uscita	15 ms	---
Totale (tempo di risposta di I/O dei data link)	94 ms	Più veloce di 32,5 ms (26% più veloce)

6-1-10 Esecuzione in background

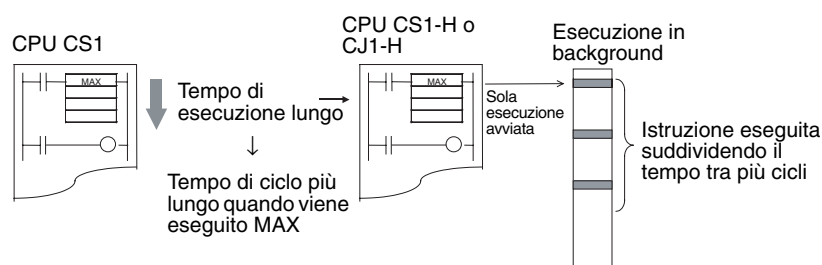
È possibile utilizzare l'esecuzione in background per ridurre le variazioni del tempo di ciclo. L'esecuzione in background è supportata solo dalle CPU CS1-H, CJ1-H o CJ1M oppure dalle CPU CS1D per sistemi a singola CPU.

L'elaborazione dei dati delle tabelle e delle stringhe di testo, ad esempio durante le ricerche di dati e stringhe di testo, richiede del tempo ed è pertanto possibile che si determinino ampie variazioni del tempo di ciclo dovute all'esecuzione di tali operazioni.

Le CPU CS1-H, CJ1-H o CJ1M o le CPU CS1D per sistemi a singola CPU (vedere nota) consentono tuttavia di eseguire in background le istruzioni riportate di seguito, suddividendo così il tempo di esecuzione tra più cicli in modo da poter controllare le variazioni del tempo di ciclo. La configurazione del PLC consente di impostare l'esecuzione in background per ciascun tipo di istruzione:

Nota Le CPU CS1D per sistemi a due CPU non supportano l'esecuzione in background.

- Istruzioni di elaborazione dei dati delle tabelle
- Istruzioni di elaborazione delle stringhe di testo
- Istruzioni di scorrimento dei dati (solo ASYNCHRONOUS SHIFT REGISTER)

L'impostazione dell'esecuzione in background delle istruzioni sopra indicate consente di controllare gli incrementi temporanei del tempo di ciclo.

Ambiti di applicazione

È possibile utilizzare l'esecuzione in background per elaborare quantità considerevoli di dati, come nel caso della compilazione, qualora tale operazione debba essere eseguita solo in determinati momenti, ad esempio una volta al giorno, e nel caso in cui sia più importante ridurre l'impatto sul tempo di ciclo che non elaborare i dati in tempi rapidi.

Procedura

1,2,3...

1. Impostare la configurazione del PLC in modo da abilitare l'esecuzione in background delle istruzioni necessarie.
2. Nella configurazione del PLC, impostare il numero della porta di comunicazione (numero logico della porta) da utilizzare per l'esecuzione in background. Il numero di porta specificato verrà utilizzato per tutte le istruzioni elaborate in background.

Nota Poiché per tutte le esecuzioni in background viene utilizzata la stessa porta, non sarà possibile eseguire un'istruzione in background se è già in corso l'esecuzione un'altra istruzione. Per controllare le istruzioni specificate per l'esecuzione in background, utilizzare il flag di abilitazione porte di comunicazione in modo da impedire l'esecuzione contemporanea di più di un'istruzione.

3. Se per un'istruzione è stata specificata l'esecuzione in background, la relativa esecuzione verrà avviata solo nel ciclo in cui vengono soddisfatte le condizioni di esecuzione e non verrà completata nel medesimo ciclo.
4. Quando viene avviata l'esecuzione in background, il flag di abilitazione porte di comunicazione relativo alla porta utilizzata viene disattivato.
5. L'esecuzione in background viene ripartita su più cicli.
6. Al termine dell'elaborazione, il flag di abilitazione porte di comunicazione relativo alla porta utilizzata viene attivato, consentendo così l'avvio dell'esecuzione in background di un'altra istruzione.

Istruzioni applicabili

■ **Istruzioni di elaborazione dei dati delle tabelle**

Istruzione	Codice mnemonico	Codice funzione
DATA SEARCH	SRCH	181
SWAP BYTES	SWAP	637
FIND MAXIMUM	MAX	182
FIND MINIMUM	MIN	183
SUM	SUM	184
FRAME CHECKSUM	FCS	180

■ **Istruzioni di elaborazione delle stringhe di testo**

Istruzione	Codice mnemonico	Codice funzione
MOVE STRING	MOV$	664
CONCATENATE STRING	+$	656
GET STRING LEFT	LEFT$	652
GET STRING RIGHT	RIGHT$	653
GET STRING MIDDLE	MID$	654
FIND IN STRING	FIND$	660
STRING LENGTH	LEN$	650
REPLACE IN STRING	RPLC$	661
DELETE STRING	DEL$	658
EXCHANGE STRING	XCHG$	665
CLEAR STRING	CLR$	666
INSERT INTO STRING	INS$	657

■ **Istruzioni di scorrimento dei dati**

Istruzione	Codice mnemonico	Codice funzione
ASYNCHRONOUS SHIFT REGISTER	ASFT	017

Differenze tra istruzioni eseguite in modalità normale e in background

Di seguito sono illustrate le differenze tra l'esecuzione normale delle istruzioni e l'esecuzione in background.

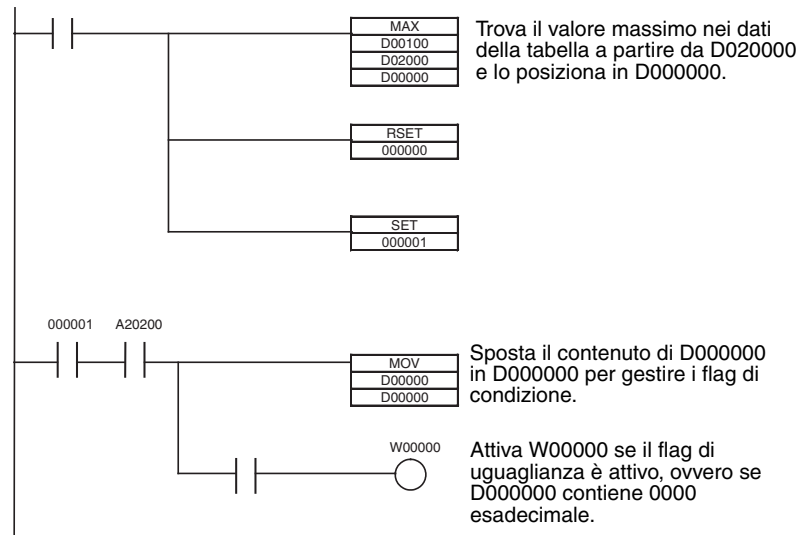
■ **Invio ai registri indice (IR)**

Se si utilizza l'istruzione MAX(182) o MIN(183) per inviare l'indirizzo di memoria I/O del canale contenente il valore minimo o massimo a un registro indice, l'indirizzo non verrà inviato al registro indice bensì ad A595 e A596. Per memorizzare l'indirizzo in un registro indice, utilizzare un'istruzione di spostamento dati, ad esempio MOVL(498), e copiare l'indirizzo in A595 e A596 in un registro indice.

■ **Flag di condizione**

I flag di condizione non vengono aggiornati a seguito dell'esecuzione di istruzioni elaborate in background. Per accedere allo stato dei flag di condizione, eseguire un'istruzione che influisca sui flag di condizione allo stesso modo, come indicato nell'esempio che segue, quindi accedere ai flag di condizione.

Ad esempio,

MOV(021) influisce sui flag di uguaglianza e sul flag negativo allo stesso modo di MAX(182), ossia entrambe le istruzioni attivano il flag di uguaglianza per 0 e attivano il flag negativo se il bit più significativo è impostato su ON. È quindi possibile utilizzare MOV(021) per copiare i risultati di MAX(182) nello stesso indirizzo e gestire i flag di condizione in modo che sia possibile accedere allo stato.

■ **Invio al registro indice IR00**

Se si utilizza l'istruzione SRCH(181) per inviare l'indirizzo di memoria I/O del canale contenente il valore corrispondente trovato (il primo canale se ne esiste più di uno) a un registro indice, l'indirizzo non verrà inviato al registro indice bensì ad A595 e A596.

■ **Invio ai registri dati (DR) per SRCH(181)**

Se si esegue SRCH(181) per inviare i dati corrispondenti trovati a un registro dati, i dati non verranno inviati al registro bensì ad A597.

■ **Stringhe di testo corrispondenti**

Se con SRCH(181) vengono trovati dati corrispondenti, non verrà attivato il flag di uguaglianza ma il flag A59801.

■ **Errori relativi alle istruzioni**

Se si verifica un errore durante l'esecuzione o un errore di accesso illegale per un'istruzione elaborata in background, non verrà attivato il flag ER o AER ma il flag A39510. A39510 resterà attivato fino alla successiva elaborazione di un'istruzione in background.

■ Invio ai registri dati (DR) per MAX(182) o MIN(183)

Se si esegue l'istruzione MAX(182) o MIN(183) specificando un registro dati come canale di uscita per il valore minimo o massimo, si verificherà un errore di esecuzione dell'istruzione e verrà attivato il flag ER.

Configurazione del PLC

Canale	Bit	Nome	Impostazione	Valore predefinito e momento di aggiornamento
198	15	Esecuzione in background di istruzioni per dati delle tabelle	0: non elaborata in background 1: elaborata in background	0: non elaborata in background Avvio dell'operazione
	14	Esecuzione in background di istruzioni per stringhe di testo	0: non elaborata in background 1: elaborata in background	
	13	Esecuzione in background di istruzioni di scorrimento dei dati	0: non elaborata in background 1: elaborata in background	
	Da 00 a 03	Numero porta di comunicazione per l'esecuzione in background	Da 0 a 7 esadecimale: porte di comunicazione da 0 a 7 (porte logiche interne)	0 esadecimale: porta 0 Avvio dell'operazione

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Descrizione
Flag di abilitazione porte di comunicazione	Da A20200 ad A20207	Viene attivato se è possibile eseguire un'istruzione di rete (SEND, RECV, CMND o PMCR) o un'operazione in background con il numero di porta corrispondente (solo CPU CS1D per sistemi a singola CPU oppure CPU CS1-H, CJ1-H o CJ1M). I bit da 00 a 07 corrispondono alle porte di comunicazione da 0 a 7. Quando si utilizza il backup semplice per eseguire un'operazione di scrittura o confronto per una schedina di memoria su una CPU CS1-H, CJ1-H o CJ1M o CS1D, viene automaticamente assegnata una porta di comunicazione e il flag corrispondente viene attivato durante l'operazione e disattivato al termine dell'operazione.
Flag di errore porte di comunicazione	Da A21900 ad A21907	Viene attivato quando si verifica un errore durante l'esecuzione di un'istruzione di rete (SEND, RECV, CMND o PMCR). I bit da 00 a 07 corrispondono alle porte di comunicazione da 0 a 7. Quando si utilizza il backup semplice per eseguire un'operazione di scrittura o confronto per una schedina di memoria su una CPU CS1-H, CJ1-H, CJ1M o CS1D, viene automaticamente assegnata una porta di comunicazione. Il flag corrispondente viene attivato se si verifica un errore e viene disattivato se l'operazione di backup semplice viene completata normalmente.

Nome	Indirizzo	Descrizione
Codici di completamento per le porte di comunicazione	Da A203 ad A210	Questi canali contengono i codici di completamento per i numeri delle porte corrispondenti al termine dell'esecuzione di un'istruzione di rete (SEND, RECV, CMND o PMCR). Il contenuto viene cancellato al termine dell'esecuzione in background (per le CPU CS1D per sistemi a singola CPU oppure per le CPU CS1-H, CJ1-H o CJ1M). I canali da A203 ad A210 corrispondono alle porte di comunicazione da 0 a 7. Quando si utilizza il backup semplice per eseguire un'operazione di scrittura o confronto per una schedina di memoria su una CPU CS1-H, CJ1-H, CJ1M o CS1D, viene automaticamente assegnata una porta di comunicazione e nel canale corrispondente viene memorizzato un codice di completamento.
Flag ER/AER per l'esecuzione in background	A39510	Viene attivato quando si verifica un errore di esecuzione o un errore di accesso illegale per un'istruzione eseguita in background. Viene disattivato all'accensione o all'avvio dell'operazione.
Uscita IR00 per l'esecuzione in background	A595 e A596	Questi canali ricevono l'uscita di un'istruzione eseguita in background quando viene specificato un registro indice come uscita. Non vi sarà alcuna uscita su IR00. Indirizzamento: da 0000 0000 a FFFF FFFF esadecimale 4 cifre inferiori: A595, 4 cifre superiori: A596
Uscita DR00 per l'esecuzione in background	A597	Questo canale riceve l'uscita di un'istruzione eseguita in background quando viene specificato un registro dati come uscita. Non vi sarà alcuna uscita su DR00. Indirizzamento: da 0000 a FFFF esadecimale
Uscita flag di uguaglianza per l'esecuzione in background	A59801	Questo flag viene attivato quando vengono trovati dati che soddisfano un'istruzione SRCH(181) eseguita in background.

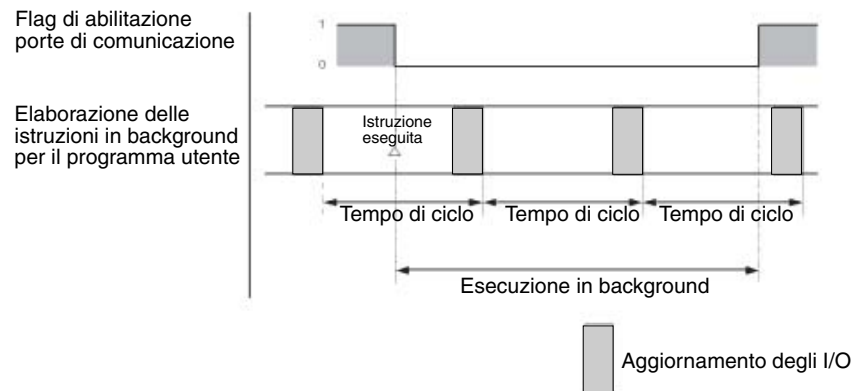
Nota Le porte di comunicazione (porte logiche interne) della CPU vengono utilizzate sia per l'esecuzione in background che per le seguenti istruzioni:

- SEND(090), RECV(098) e CMND(490) (istruzioni di comunicazione in rete)
- PMCR(260) (PROTOCOL MACRO)

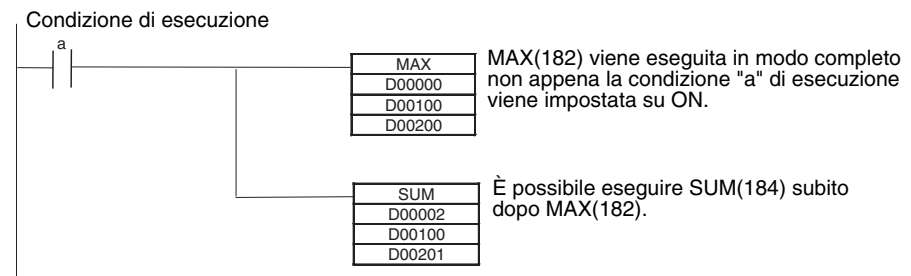
Non è possibile eseguire contemporaneamente sulla stessa porta tali istruzioni e le istruzioni in background. Per accertarsi che venga eseguita una sola istruzione per volta su ciascuna porta, utilizzare i flag di abilitazione porte di comunicazione.

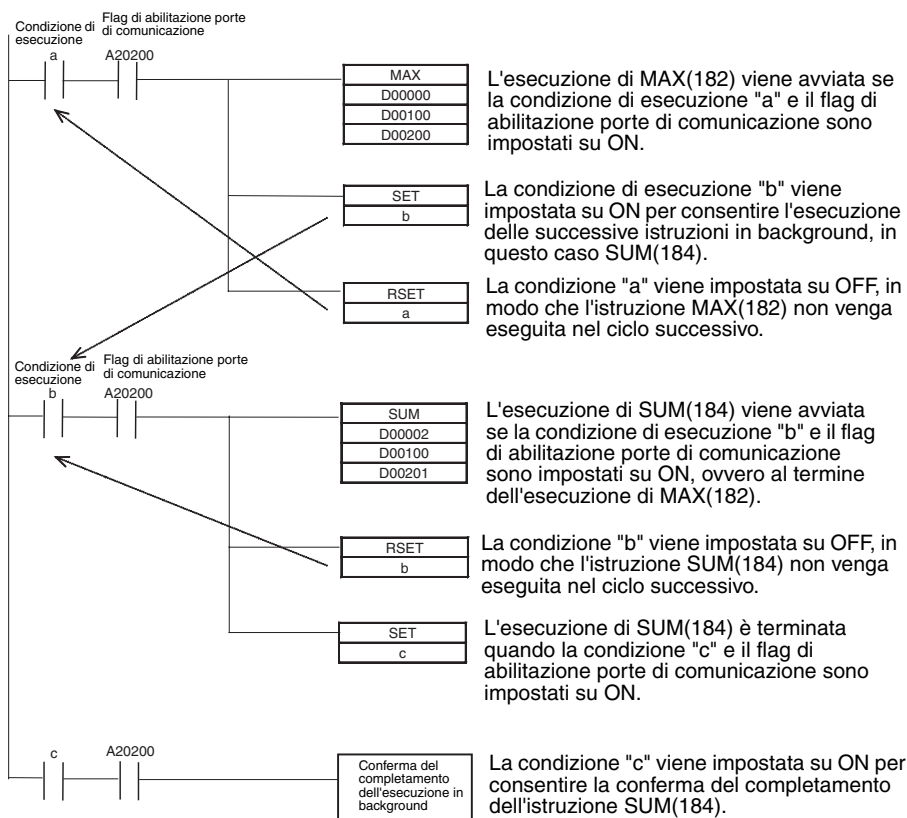
Nota Se si specifica un'istruzione per l'esecuzione in background su una porta il cui flag di abilitazione porte di comunicazione è impostato su OFF, il flag ER verrà impostato su ON e l'istruzione in background non verrà eseguita.

Flag di abilitazione porte di comunicazione

I flag di abilitazione porte di comunicazione vengono attivati quando la porta non è utilizzata e disattivati quando sulla porta è in corso un'elaborazione.

Esempio di programmazione 1

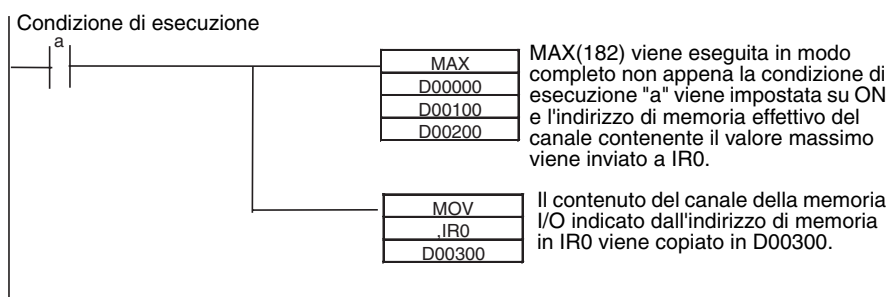
■ Programmazione normale senza esecuzione in background

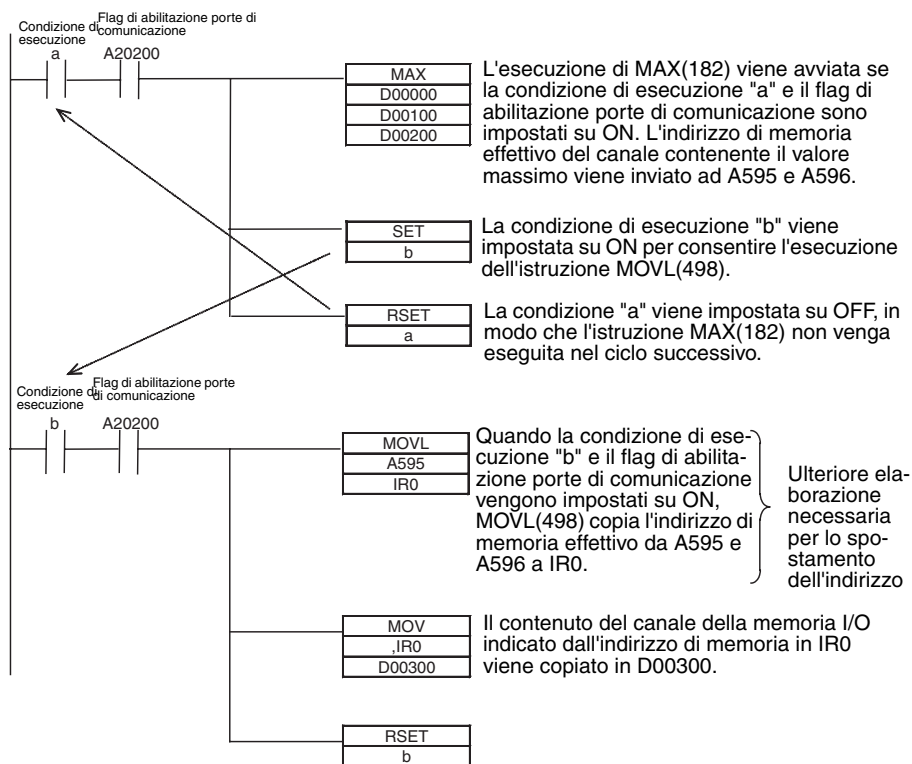
Come illustrato di seguito, l'elaborazione viene completata quando viene eseguita l'istruzione.

■ Programmazione con esecuzione in background

Nel caso dell'esecuzione in background, il programma viene modificato in modo che l'istruzione MAX(182) venga eseguita solo quando il flag di abilitazione per la porta di comunicazione specificata è attivato, ovvero solo quando la porta non è utilizzata per altre esecuzioni in background o comunicazioni di rete. Inoltre, le condizioni di ingresso vengono controllate mediante le istruzioni SET e RESET per garantire l'esecuzione dell'elaborazione nell'ordine corretto. Nell'esempio che segue la porta di comunicazione 0 viene utilizzata per l'esecuzione in background.

Esempio di programmazione 2

In questo esempio viene illustrata l'esecuzione in background quando viene specificato un registro indice come uscita, possibile con le istruzioni MAX(182), MIN(183) e SRCH(181).

■ Programmazione normale senza esecuzione in background

Come illustrato di seguito, l'indirizzo di memoria effettivo del canale contenente il valore massimo viene inviato a un registro indice.

■ Programmazione con esecuzione in background

Nel caso dell'esecuzione in background, l'indirizzo di memoria effettivo del canale contenente il valore massimo viene inviato ad A595 e A596. Viene quindi utilizzata l'istruzione MOVL(498) per spostare l'indirizzo di memoria effettivo nel registro indice.

6-1-11 Condivisione dei registri indice e dati tra i task

La condivisione di registri indice e registri dati (IR/DR) tra i task è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D. L'impostazione normale prevede registri separati per ciascun task. È possibile confermare l'impostazione corrente in A09914.

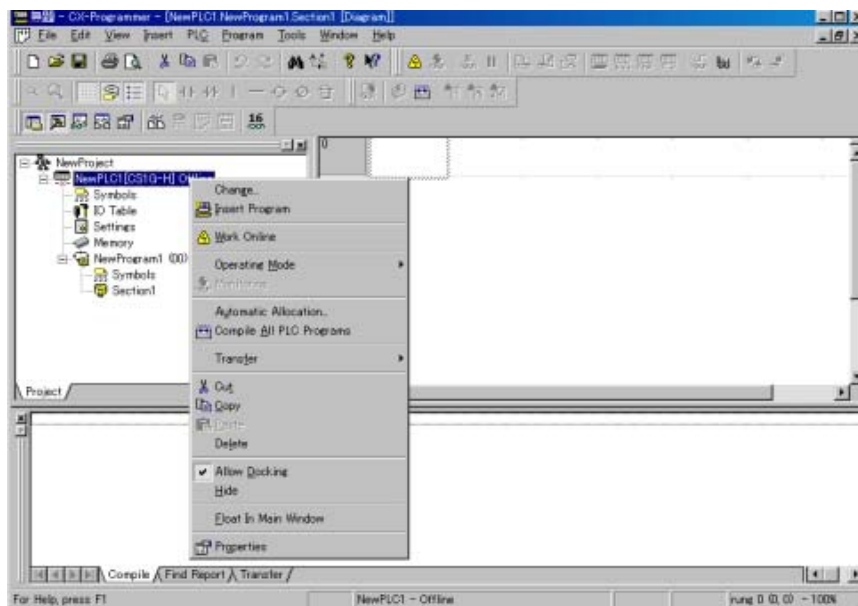
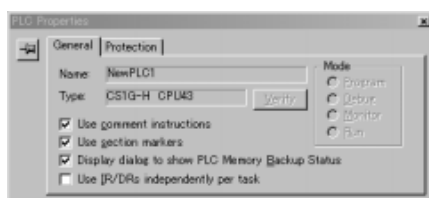
Nota

1. È possibile utilizzare registri indice e dati condivisi per eliminare la necessità di memorizzare e caricare il contenuto di un registro nei task quando due o più task richiedono il medesimo contenuto. Per ulteriori informazioni sulla memorizzazione e sul caricamento del contenuto dei registri indice, fare riferimento al capitolo dedicato ai registri indice nel manuale *CS Series Operation Manual* (W339) o nel *Manuale dell'operatore della serie CJ* (W393).
2. La condivisione di registri indice e dati consente di ridurre i tempi di passaggio da un task al successivo. È consigliabile impostare la condivisione dei registri se i registri non sono in uso oppure quando non esiste un motivo particolare per mantenere separati i registri nei vari task.

Metodo di impostazione

Utilizzare CX-Programmer per impostare registri indice e dati condivisi. Non è possibile eseguire questa operazione da una Console di programmazione.

1,2,3...

1. Selezionare un PLC nella struttura dei progetti di CX-Programmer e fare clic con il pulsante destro del mouse su di esso.

2. Selezionare ***Properties*** (Proprietà). Viene visualizzata la seguente finestra di dialogo.

3. Se per ciascun task sono necessari registri indice e dati separati, lasciare selezionata l'opzione per l'uso indipendente di IR/DR. Per utilizzare registri indice e dati condivisi per tutti i task, deselezionare l'opzione.

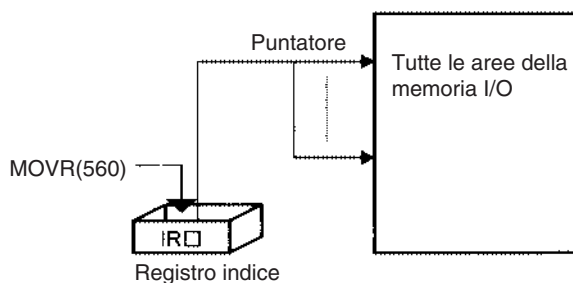
Flag e canali dell'area ausiliaria

Nome	Indirizzo	Descrizione
Condivisione IR/DR tra task	A09914	Indica se i registri indice e dati sono condivisi o meno tra i task. 0: registri separati per ciascun task (impostazione predefinita) 1: registri condivisi per tutti i task

6-2 Registri indice

6-2-1 Descrizione dei registri indice

I registri indice svolgono la funzione di puntatori per specificare indirizzi di memoria del PLC, i quali rappresentano indirizzi di memoria assoluti nella memoria I/O. Dopo avere memorizzato gli indirizzi di memoria del PLC in un registro indice tramite l'istruzione MOVR(560) o MOVRW(561), è possibile inserire il registro indice come operando in altre istruzioni per specificare in modo indiretto l'indirizzo di memoria del PLC.

Il vantaggio offerto dai registri indice consiste nella possibilità di specificare bit o canali della memoria I/O, compresi i valori attuali di temporizzatore e contatore.

6-2-2 Utilizzo dei registri indice

I registri indice sono uno strumento molto potente se utilizzati in combinazione con cicli di tipo FOR-NEXT. È possibile incrementare, decrementare ed eseguire facilmente l'offset del contenuto dei registri indice, in modo che un numero ridotto di istruzioni in un ciclo consenta di elaborare efficacemente tabelle di dati consecutivi.

Funzionamento di base

L'utilizzo dei registri indice si basa sostanzialmente sulla procedura descritta di seguito.

1,2,3...

1. Utilizzare MOVR(560) per memorizzare l'indirizzo di memoria del PLC relativo al bit o al canale desiderato in un registro indice.
2. Definire il registro indice come operando in quasi tutte le istruzioni per specificare l'indirizzo del bit o del canale desiderato in modo indiretto.
3. Definire l'offset o l'incremento dell'indirizzo di memoria originale del PLC (vedere di seguito) per reindirizzare il puntatore a un altro indirizzo.
4. Ripetere i punti 2 e 3 per eseguire l'istruzione su qualsiasi numero di indirizzi.

Offset, incremento e decremento degli indirizzi

Nella seguente tabella sono riportate le variazioni disponibili per l'indirizzamento indiretto.

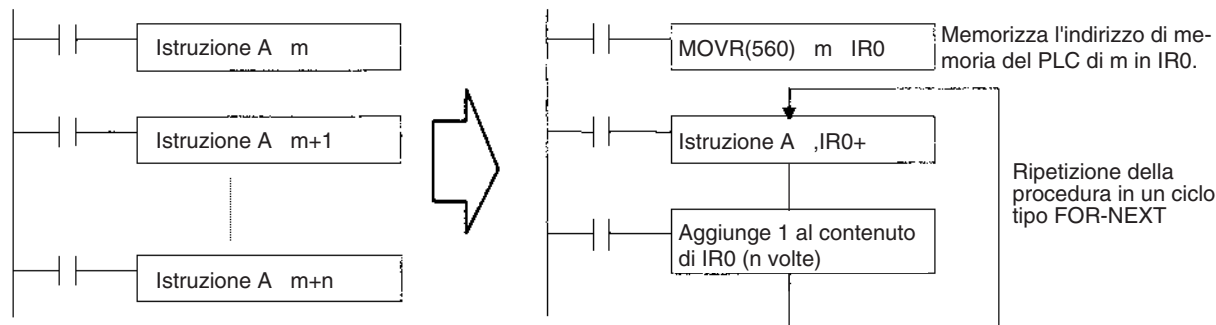
Variazione	Sintassi
Indirizzamento indiretto	,IR□
Indirizzamento indiretto con offset costante	Costante, IR□ (includere un segno più (+) o un segno meno (-) nella costante)
Indirizzamento indiretto con offset DR	DR□,IR□
Indirizzamento indiretto con incremento automatico	Incremento di 1: ,IR□+ Incremento di 2: ,IR□++
Indirizzamento indiretto con decremento automatico	Decremento di 1: ,–IR□ Decremento di 2: ,– –IR□

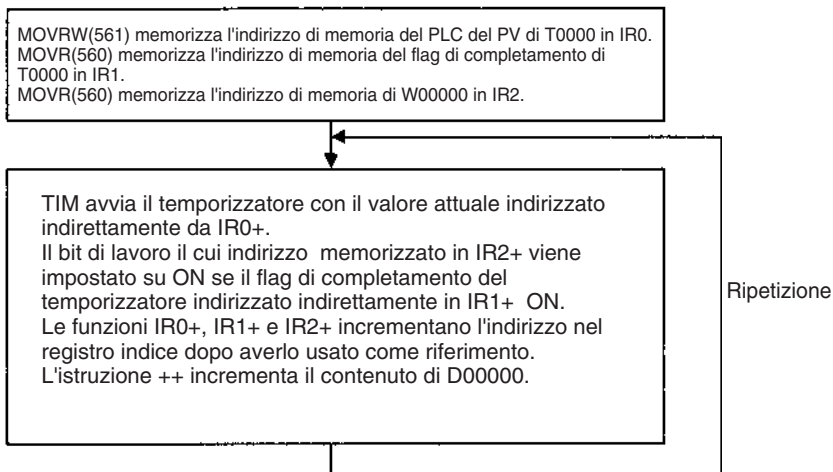
Istruzioni per l'indirizzamento diretto dei registri indice

È possibile indirizzare in modo diretto i registri indice tramite le istruzioni riportate di seguito.

DOUBLE SIGNED BINARY ADD WITHOUT CARRY: +L(401), DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY: –L(411), DOUBLE INCREMENT BINARY: ++L(591) e DOUBLE DECREMENT BINARY: – –L(593)

Esempio 1

L'esempio seguente illustra in che modo è possibile sostituire un registro indice a una lunga serie di istruzioni in un ciclo di programma. In questo caso, l'istruzione A viene ripetuta n+1 volte per l'esecuzione di operazioni quali la lettura e il confronto di una tabella di valori.

Esempio 2

Nell'esempio seguente i registri indice vengono utilizzati in un ciclo FOR–NEXT per definire e avviare 100 temporizzatori (da T0000 a T099) con i valori impostati presenti nell'intervallo da D00100 a D00109. Il numero di temporizzatore e il flag di completamento di ciascun temporizzatore sono specificati nei registri indice e il ciclo viene ripetuto man mano che i registri indice vengono incrementati di uno a ciascuna ripetizione.

La subroutine di 11 istruzioni a sinistra equivale alla subroutine di 200 istruzioni illustrata a destra.

Indirizzamento diretto dei registri indice

È possibile indirizzare i registri di indici in modo diretto solo nelle istruzioni riportate nella tabella seguente.

Gruppo di istruzioni	Nome dell'istruzione	Codice mnemonico	Funzione primaria
Istruzioni di spostamento dati	MOVE TO REGISTER	MOVR(560)	Memorizza l'indirizzo di memoria del PLC relativo a un bit o canale in un registro indice.
	MOVE TIMER/COUNTER PV TO REGISTER	MOVRW(561)	
Istruzioni di elaborazione dei dati delle tabelle	SET RECORD LOCATION	SETR(635)	
	GET RECORD NUMBER	GETR(636)	Invia l'indirizzo di memoria del PLC memorizzato a un registro indice.
Istruzioni di spostamento dati	DOUBLE MOVE	MOVL(498)	Esegue trasferimenti tra i registri indice. Viene utilizzata per scambi e confronti.
	DOUBLE DATA EXCHANGE	XCGL(562)	
Istruzioni di confronto	DOUBLE EQUAL	=L(301)	
	DOUBLE NOT EQUAL	< >L(306)	
	DOUBLE LESS THAN	< L(311)	
	DOUBLE LESS THAN OR EQUAL	< =L(316)	
	DOUBLE GREATER THAN	>L(321)	
	DOUBLE GREATER THAN OR EQUAL	>=L(326)	
	DOUBLE COMPARE	CMPL(060)	
Istruzioni di incremento e decremento	DOUBLE INCREMENT BINARY	++L(591)	Modifica l'indirizzo di memoria del PLC nel registro indice incrementando, decrementando o eseguendo l'offset del contenuto.
	DOUBLE DECREMENT BINARY	– –L(593)	
Istruzioni matematiche dei simboli	DOUBLE SIGNED BINARY ADD WITHOUT CARRY	+L(401)	
	DOUBLE SIGNED BINARY SUBTRACT WITHOUT CARRY	–L(411)	
Istruzioni speciali	CONVERT ADDRESS FROM CV	FRMCV(284)	Converte gli indirizzi di memoria effettivi dei PLC da indirizzi per la serie CV in indirizzi per la serie CS/CJ. Solo CPU CS1-H, CJ1-H, CJ1M o CS1D.
	CONVERT ADDRESS TO CV	TOCV(285)	

Nota Le istruzioni per operandi a lunghezza doppia (ovvero quelli con una "L" finale) vengono utilizzate per i registri indice da IR0 a IR15 in quanto ciascun registro contiene due canali.

6-2-3 Elaborazione correlata ai registri indice

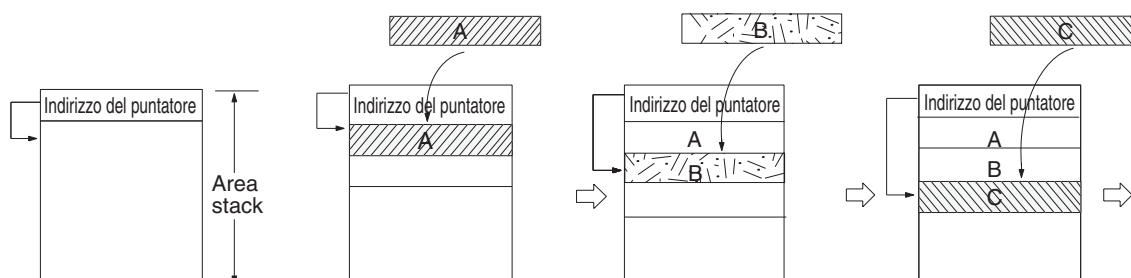
Le istruzioni di elaborazione dei dati delle tabelle delle CPU della serie CS/CJ sono complementari alle funzioni dei registri indice. È possibile suddividere tali istruzioni in istruzioni per l'elaborazione degli stack e istruzioni per l'elaborazione delle tabelle.

Elaborazione	Funzione	Istruzioni
Elaborazione di stack	Elaborazione di tabelle di dati FIFO (First-In First-Out) o LIFO (Last-In First-Out) e lettura, scrittura, inserimento, eliminazione o conteggio delle voci di dati nelle tabelle di dati	SSET(630), PUSH(632), FIFO(633), LIFO(634) e, solo per le CPU CS1-H, CJ1-H, CJ1M o CS1D, SREAD(639), SWRITE(640), SINS(641), SDEL(642), SNUM(638)

Elaborazione			Funzione	Istruzioni
Elaborazione di tabelle	Tabelle con record a un solo canale (istruzioni su intervalli di valori)	Elaborazione di base	Individuazione di valori, quali la checksum, un valore specifico, il valore massimo o minimo, all'interno della gamma	FCS(180), SRCH(181), MAX(182), MIN(183) e SUM(184)
		Elaborazioni speciali	Altri tipi di elaborazione delle tabelle, ad esempio confronto o ordinamento	Combinazione dei registri indice con istruzioni quali SRCH(181), MAX(182), MIN(183) e istruzioni di confronto
	Tabelle con record a più canali (istruzioni per tabelle di record)		Elaborazione di dati organizzati in record di lunghezza pari a più canali	Combinazione dei registri indice con istruzioni quali DIM(631), SETR(635), GETR(636) e istruzioni di confronto

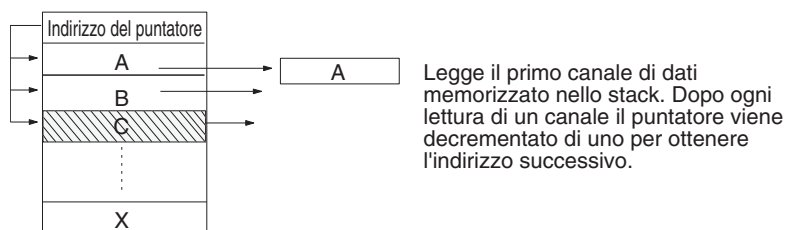
Elaborazione dello stack

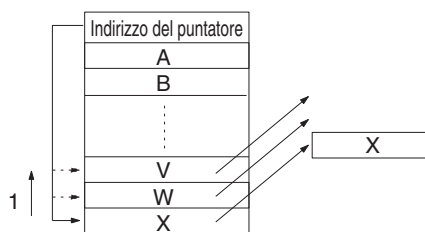
Le istruzioni di stack vengono utilizzate per tabelle di dati definite in modo specifico denominate stack. È possibile estrarre dati da uno stack in base a un criterio FIFO (First-In First-Out, primo a entrare primo a uscire) o LIFO (Last-In First-Out, ultimo a entrare primo a uscire).

È necessario definire una determinata area della memoria I/O come stack. I primi canali dello stack indicano la lunghezza dello stack e contengono il puntatore dello stack. Il puntatore dello stack viene incrementato ogni volta che vengono scritti dati nello stack, ad indicare l'indirizzo successivo in cui dovranno essere memorizzati i dati.

Lo schema precedente illustra lo stato dei dati del puntatore prima dell'aggiunta di dati.

Nota I primi due canali dello stack contengono l'indirizzo di memoria del PLC relativo all'ultimo canale dello stack, mentre il canale successivo contiene il puntatore dello stack.

Elaborazione FIFO (First-In First-Out)

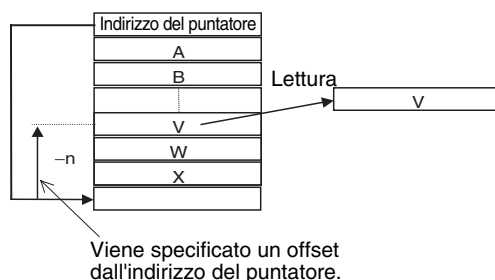
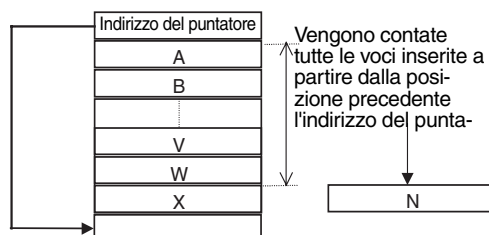
Lo schema riportato di seguito illustra il funzionamento di uno stack FIFO.

Elaborazione LIFO (Last-In First-Out)

Lo schema riportato di seguito illustra il funzionamento di uno stack LIFO.

Legge l'ultimo canale di dati memorizzato nello stack. Dopo ogni lettura di un canale il puntatore viene decrementato di uno per ottenere l'indirizzo successivo.
I dati nella posizione appena letta rimangono invariati.

Gestione di dati specifici delle tabelle

È possibile leggere, scrivere, inserire o eliminare singole voci in una tabella. Lo schema riportato di seguito illustra un esempio di lettura.

Conteggio dei dati delle tabelle

Nello schema che segue viene illustrato il conteggio dei dati in una tabella.

Il numero di voci inserite in una tabella viene contato a partire dalla posizione immediatamente precedente all'indirizzo del puntatore verso l'inizio della tabella.
È possibile utilizzare questa funzione per contare, ad esempio, il numero di oggetti su un nastro trasportatore.

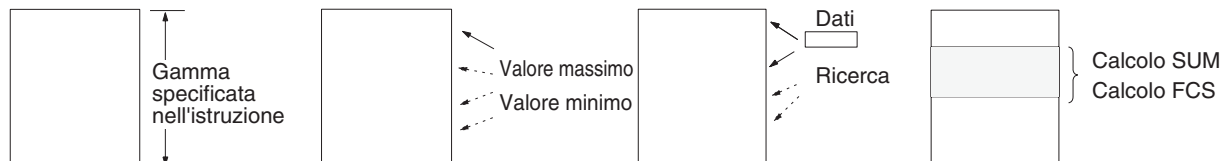
Istruzioni di stack

Nella tabella riportata di seguito sono elencate le istruzioni di stack e le relative funzioni. Ambiti di applicazione tipici in cui vengono utilizzati gli stack sono l'elaborazione delle informazioni relative agli scaffali nei sistemi di stoccaggio automatizzato, l'elaborazione dei risultati dei test e la gestione delle informazioni relative agli oggetti su un nastro trasportatore.

Istruzione	Funzione
SSET(630)	Definisce un'area stack.
PUSH(632)	Memorizza i dati nel canale successivo disponibile nello stack.
FIFO(633)	Legge i dati dallo stack in base a un criterio "first-in first-out" (primo a entrare primo a uscire).
LIFO(634)	Legge i dati dallo stack in base a un criterio "last-in first-out" (ultimo a entrare primo a uscire).
SREAD(639)	Legge una voce specifica dalla tabella (solo CPU CS1-H, CJ1-H, CJ1M o CS1D).
SWRITE(640)	Scrive una voce specifica nella tabella (solo CPU CS1-H, CJ1-H, CJ1M o CS1D).
SINS(641)	Inserisce una voce specifica nella tabella (solo CPU CS1-H, CJ1-H, CJ1M o CS1D).
SDEL(642)	Elimina una voce specifica dalla tabella (solo CPU CS1-H, CJ1-H, CJ1M o CS1D).
SNUM(638)	Conta il numero di voci inserite nella tabella (solo CPU CS1-H, CJ1-H, CJ1M o CS1D).

Elaborazione delle tabelle (istruzioni su intervalli di valori)

Le istruzioni su intervalli di valori agiscono su un intervallo di canali, che può essere considerato una tabella di record a un solo canale. Tali istruzioni eseguono operazioni di base quali il recupero dei valori minimo e massimo all'interno della gamma, la ricerca di un determinato valore o il calcolo della somma o dell'FCS (checksum dei frame).

L'indirizzo di memoria del PLC relativo al canale del risultato (il canale contenente il valore massimo o minimo, i dati trovati e così via) viene automaticamente memorizzato in IR0. È possibile utilizzare il registro indice (IR0) come operando in istruzioni successive, ad esempio MOV(021), per la lettura del contenuto del canale o l'esecuzione di ulteriori elaborazioni.

Nella tabella riportata di seguito sono elencate le istruzioni su intervalli di valori e le relative funzioni.

Istruzione	Funzione	Descrizione
SRCH(181)	Trova i dati cercati.	Trova i dati cercati all'interno della gamma specificata e invia a IR0 l'indirizzo di memoria del PLC relativo al canale che contiene tale valore.
MAX(182)	Trova il valore massimo.	Trova il valore massimo all'interno della gamma specificata e invia a IR0 l'indirizzo di memoria del PLC relativo al canale che contiene tale valore.
MIN(183)	Trova il valore minimo.	Trova il valore minimo all'interno della gamma specificata e invia a IR0 l'indirizzo di memoria del PLC relativo al canale che contiene tale valore.
SUM(184)	Calcola la somma.	Calcola la somma dei dati nella gamma specificata.
FCS(180)	Calcola il valore di checksum.	Calcola il valore di checksum dei frame per i dati nella gamma specificata.

È possibile combinare i registri indice con altre istruzioni, ad esempio le istruzioni di confronto, in ciclo FOR–NEXT per l'esecuzione di operazioni più complesse su gamme di canali.

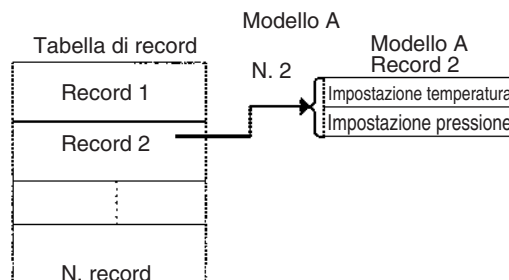
Elaborazione delle tabelle (istruzioni per le tabelle di record)

Le istruzioni per le tabelle di record vengono utilizzate per tabelle di dati definite in modo specifico costituite da record di uguale lunghezza. È possibile accedere ai record in base al numero di record per semplificare l'elaborazione.

Istruzione	Funzione	Descrizione
DIM(631)	Definisce una tabella di record.	Dichiara la lunghezza di ciascun record e il numero dei record.
SETR(635)	Imposta la posizione del record.	Scrive la posizione del record specificato (l'indirizzo di memoria del PLC relativo all'inizio del record) nel registro indice specificato.
GETR(636)	Recupera la posizione del record.	Restituisce il numero di record del record contenente l'indirizzo di memoria del PLC nel registro indice specificato.

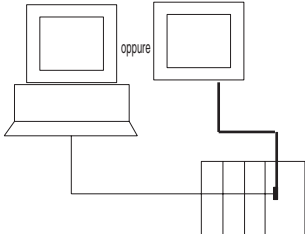
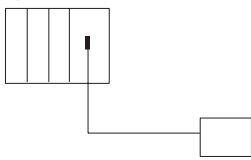
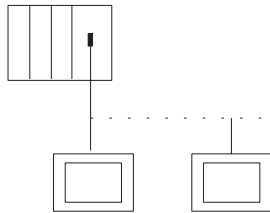
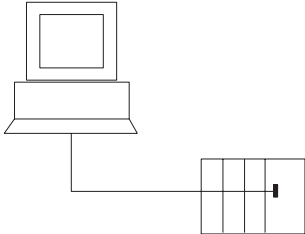
Nota I numeri di record e gli indirizzi dei canali vengono messi in relazione tramite i registri indice. Specificare un numero di record in SETR(635) per memorizzare l'indirizzo di memoria del PLC relativo all'inizio del record in un registro indice. Quando è necessario recuperare dati presenti in tale record, aggiungere l'offset appropriato al registro indice per accedere a qualsiasi canale del record.

Utilizzare le istruzioni per la tabella di record con i registri indice per i seguenti tipi di operazione: lettura/scrittura dei dati dei record, ricerca di record, ordinamento dei dati dei record, confronto dei dati dei record ed esecuzione di calcoli sui dati dei record.

Un'applicazione tipica delle tabelle di record è rappresentata dalla memorizzazione dei dati di produzione per differenti modelli di un prodotto, ad esempio le impostazioni di temperatura e pressione, sotto forma di record e il passaggio da un modello all'altro grazie alla sola modifica del numero di record.

L'utilizzo delle tabelle di record si basa sostanzialmente sulla procedura descritta di seguito.

1,2,3...

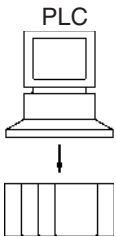
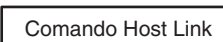
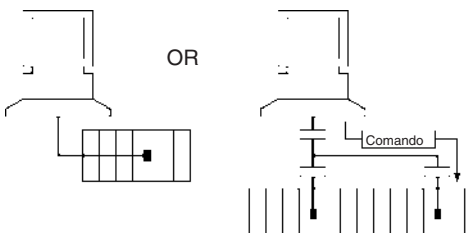
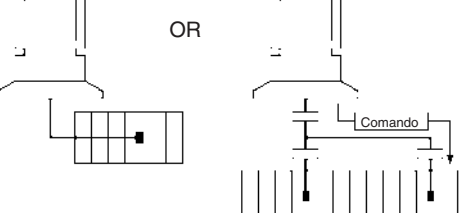
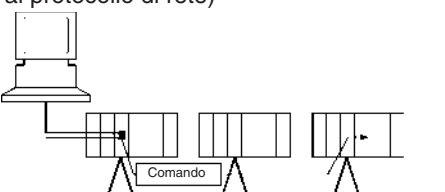
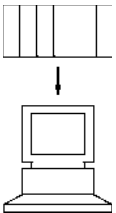
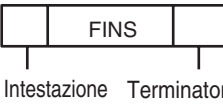
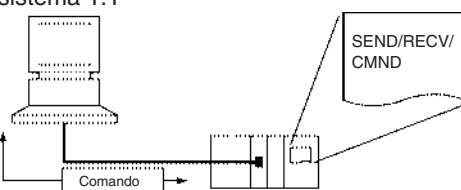
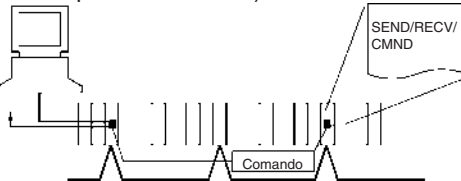
1. Definire la struttura della tabella di record mediante l'istruzione DIM(631) e impostare l'indirizzo di memoria del PLC relativo a un record in un registro indice tramite SETR(635).
2. Eseguire l'offset o incrementare l'indirizzo di memoria del PLC nel registro indice per leggere o confrontare canali all'interno del record.
3. Eseguire l'offset o incrementare l'indirizzo di memoria del PLC nel registro indice per passare a un altro record.
4. Ripetere le operazioni delle fasi 2 e 3 in base alle esigenze.

Esempio

Nell'esempio riportato di seguito i registri indice e le istruzioni per le tabelle di record vengono utilizzati per confrontare tre valori rispetto ai canali 1, 3 e 5 di ciascun record. Quando viene rilevata una corrispondenza, il numero di record viene memorizzato in D00000.

0000
01
DIM
1
&5
&1000
E0_00000
Definisce la tabella di record 1 contenente 1000 record con 5 canali ciascuno.
SETR
1
&0
IR0
Memorizza l'indirizzo di memoria del PLC del primo record della tabella n. 1 (record 0) in IR0.
JMP
&1000
Passa oltre il ciclo FOR-NEXT se le condizioni di esecuzione non sono state impostate.
FOR
&1000
Controlla il ciclo FOR-NEXT fino a un massimo di 1000 ripetizioni (1000 record).
=
,IR0
#1234
=
+2,IR0
#ABCD
=
+4,IR0
#9999
GETR
&1
IR0
D00000
BREAK
Confronta il contenuto del canale 1 con il valore 1234, del canale 3 con il valore ABCD e del canale 5 con il valore 9999.
Se i canali 1, 3 e 5 contengono i dati di confronto, il numero del record viene memorizzato in D00000 e il ciclo viene interrotto eseguendo l'istruzione BREAK.
ON
+L
IR0
&5
IR0
Aggiunge 5 all'indirizzo di memoria del PLC in IR0 per spostarsi all'inizio del record successivo.
NEXT
L'esecuzione del ciclo riprende dall'istruzione FOR.
JME
&1000
Salto alla destinazione specificata se le condizioni di esecuzione non sono state impostate.
Incrementa il valore in IR0 di 5 ad ogni ripetizione e confronta il primo, terzo e quinto canale di ogni record ai dati di confronto. Scrive il numero del record in D00000 e interrompe il ciclo se trova una corrispondenza con i dati.

6-3 Comunicazione seriale

Le CPU della serie CS/CJ supportano le funzioni di comunicazione seriale indicate di seguito. La comunicazione Host Link e la comunicazione senza protocollo verranno descritte in dettaglio più avanti in questa sezione.

Protocollo	Collegamenti	Descrizione	Porte	
			Periferica	RS-232C
Host Link	Computer host / Terminale OMRON (oppure)	1) È possibile eseguire vari comandi di controllo, come la lettura e la scrittura della memoria I/O, il cambio di modalità operativa e l'impostazione o il ripristino forzato dei bit, inviando comandi Host Link o FINS dal computer host alla CPU. 2) È inoltre possibile inviare comandi FINS dalla CPU al computer host per trasmettere dati o informazioni. Utilizzare le comunicazioni Host Link per monitorare dati quali lo stato operativo, informazioni sugli errori e dati di qualità a livello di PLC, o inviare dati al PLC, ad esempio informazioni di pianificazione della produzione.	OK	OK
Senza protocollo	Dispositivo esterno standard	Consente di comunicare mediante dispositivi standard collegati alla porta RS-232C senza un formato comando-risposta. Per trasmettere i dati dalla porta di trasmissione o leggere i dati dalla porta di ricezione, vengono eseguite le istruzioni TXD(236) e RXD(235) dal programma. È possibile specificare intestazioni di frame e codici di fine.	Non consentita	OK
NT link 1:N o 1:1	Terminali OMRON	È possibile scambiare dati con terminali programmabili senza utilizzare un programma di comunicazione nella CPU.	OK	OK
Bus di periferica	Dispositivi di programmazione (Console di programmazione escluse)	Consente comunicazioni ad alta velocità con dispositivi di programmazione diversi dalle Console di programmazione. La programmazione remota tramite modem non è supportata.	OK	OK

Protocollo	Collegamenti	Descrizione	Porte	
			Periferica	RS-232C
Gateway seriale (conversione in CompoWay/F) (solo CPU versione 3.0 o successiva)	Componenti OMRON (compatibili con dispositivi CompoWay/F) Terminale della serie NS FINS PLC Gateway seriale CompoWay/F Percorso di comunicazione seriale Componenti OMRON	Conversione dei comandi FINS ricevuti in comandi CompoWay/F e trasferimento di tali comandi sul percorso di comunicazione seriale.	OK	OK
Collegamenti seriali tra PLC (solo CJ1M)	CPU CJ1M Modulo di polling CJ1W-CIF11 collegato alla porta RS-232C (vedere nota) RS-422A/485 Per terminale della serie NS: NS-AL002 CPU CJ1M Modulo sottoposto a polling CPU CJ1M Modulo sottoposto a polling 8 Moduli al massimo CPU CJ1M Modulo di polling RS-232C CPU CJ1M Modulo sottoposto a polling	È possibile condividere fino a 10 canali per Modulo per un massimo di 9 CPU, inclusi un Modulo di polling e otto Moduli sottoposti a polling. È possibile collegare un adattatore RS-422A alla porta RS-232C di ciascuna CPU per la comunicazione via RS-422A/485. In alternativa, è possibile stabilire la comunicazione tra due CPU tramite un collegamento RS-232C. I collegamenti seriali tra PLC possono comprendere anche terminali programmabili come unità sottoposte a polling attraverso una connessione NT Link (1:N) in combinazione con CPU CJ1M.	Non consentita	OK

Di seguito sono descritti i tipi di comunicazione Host Link e senza protocollo.

Nota L'adattatore CJ1W-CIF11 non dispone di isolamento e la distanza di trasmissione totale massima è di 50 metri. Se la distanza di trasmissione totale è maggiore di 50 metri, utilizzare l'adattatore NT-AL001 isolato anziché l'adattatore CJ1W-CIF11. Se si utilizza solo l'adattatore NT-AL001, la distanza di trasmissione totale massima sarà pari a 500 metri.

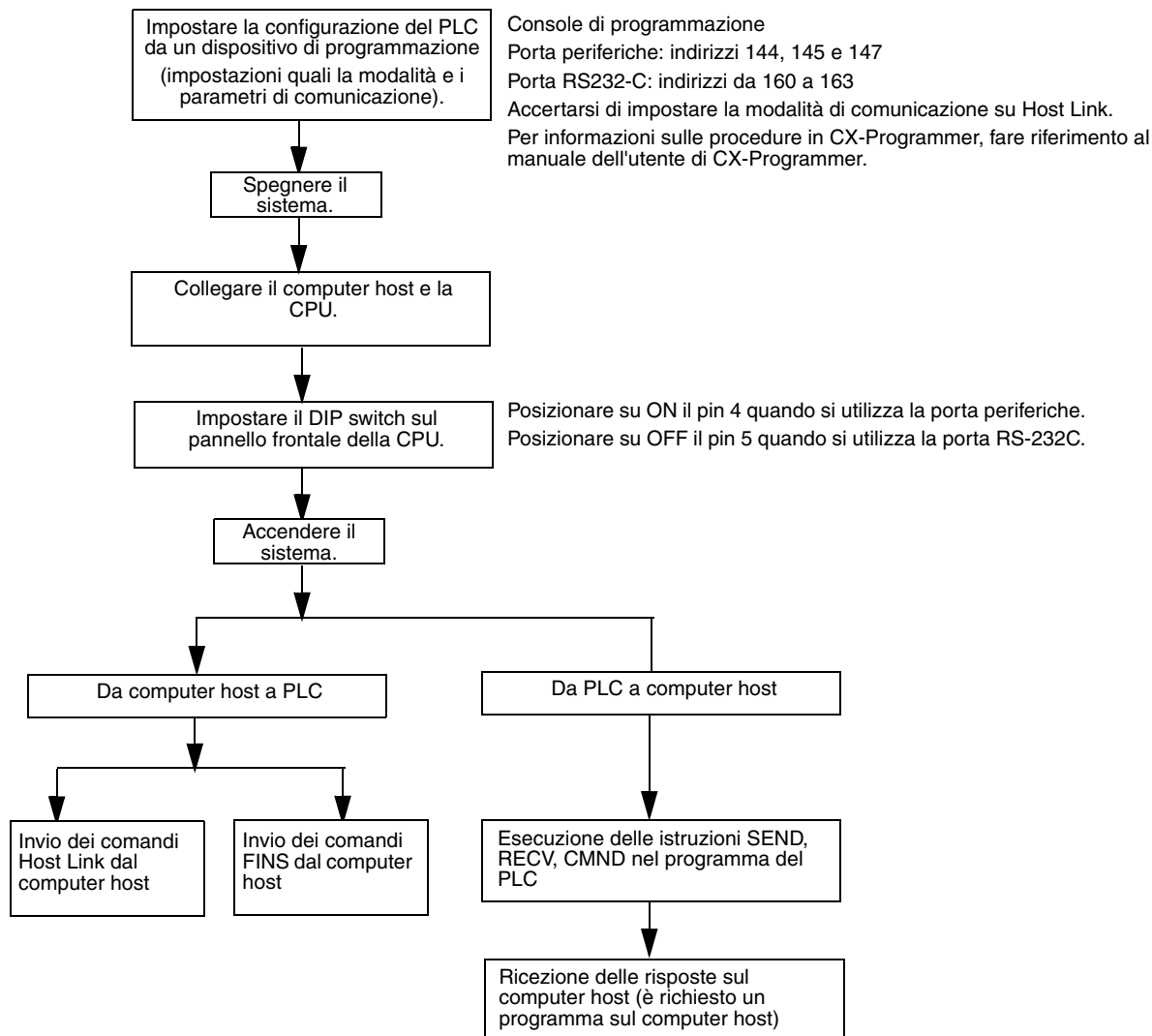
6-3-1 Comunicazione Host Link

Nella tabella seguente sono riportate le funzioni di comunicazione Host Link disponibili nei PLC della serie CS/CJ. Scegliere il metodo che meglio si adatta alla propria applicazione.

Flusso dei comandi	Tipo di comando	Metodo di comunicazione	Configurazione
Computer host PLC	Comando Host Link Comando Host Link	Creazione di frame nel computer host e invio del comando al PLC. Ricezione della risposta dal PLC. Applicazione: utilizzare questo metodo quando la comunicazione ha luogo principalmente dal computer host al PLC.	Connessione diretta al computer host in un sistema 1:1 o 1:N OR Comando
	Comando FINS1 (con un'intestazione e un terminatore Host Link) FINS Intestazione Terminatore	Creazione di frame nel computer host e invio del comando al PLC. Ricezione della risposta dal PLC. Applicazione: utilizzare questi metodi quando la comunicazione ha luogo principalmente dal computer host a PLC in rete.	Connessione diretta al computer host in un sistema 1:1 o 1:N OR Comando
			Comunicazione dal computer host con altri PLC nella rete (conversione da Host Link al protocollo di rete) Comando
PLC Computer host	Comando FINS2 (con un'intestazione e un terminatore Host Link) FINS Intestazione Terminatore	Invio di frame tramite le istruzioni SEND, RECV e CMND della CPU. Ricezione della risposta dal computer host. Applicazione: utilizzare questo metodo quando la comunicazione ha luogo principalmente dal PLC al computer host per la trasmissione di informazioni sullo stato, ad esempio le informazioni sugli errori.	Connessione diretta al computer host in un sistema 1:1 SEND/RECV/ CMND Comando
			Comunicazione con il computer host tramite altri PLC nella rete (conversione da Host Link al protocollo di rete) SEND/RECV/ CMND Comando

Nota

1. È necessario associare al comando FINS un'intestazione e un terminatore Host Link prima di trasmetterlo dal computer host.
2. Il comando FINS viene trasmesso dal PLC con un'intestazione e un terminatore Host Link associato. È necessario predisporre un programma sul computer host per l'analisi dei comandi FINS e la restituzione delle risposte appropriate.

Procedura

Comandi Host Link

Nella tabella riportata di seguito sono elencati i comandi Host Link. Per ulteriori informazioni, fare riferimento al manuale *C-series Host Link Units System Manual* (W143).

Codice intesta-zione	Nome	Funzione
RR	CIO AREA READ	Legge il contenuto del numero specificato di canali dell'area CIO, a partire dal canale specificato.
RL	LINK AREA READ	Legge il contenuto del numero specificato di canali dell'area dei data link, a partire dal canale specificato.
RH	HR AREA READ	Legge il contenuto del numero specificato di canali dell'area di ritenzione, a partire dal canale specificato.
RC	PV READ	Legge il contenuto del numero specificato di valori attuali dei temporizzatori/contatori, a partire dal temporizzatore/contatore specificato.
RG	T/C STATUS READ	Legge lo stato dei flag di completamento del numero specificato di temporizzatori/contatori, a partire dal temporizzatore/contatore specificato.
RD	DM AREA READ	Legge il contenuto del numero specificato di canali dell'area DM, a partire dal canale specificato.
RJ	AR AREA READ	Legge il contenuto del numero specificato di canali dell'area ausiliaria, a partire dal canale specificato.
RE	EM AREA READ	Legge il contenuto del numero specificato di canali dell'area EM, a partire dal canale specificato.

Codice intesta-zione	Nome	Funzione
WR	CIO AREA WRITE	Scrive i dati specificati (solo unità canale) nell'area CIO, a partire dal canale specificato.
WL	LINK AREA WRITE	Scrive i dati specificati (solo unità canale) nell'area dei data link, a partire dal canale specificato.
WH	HR AREA WRITE	Scrive i dati specificati (solo unità canale) nell'area di ritenzione, a partire dal canale specificato.
WC	PV WRITE	Scrive i valori attuali del numero specificato di temporizzatori/contatori, a partire dal temporizzatore/contatore specificato.
WD	DM AREA WRITE	Scrive i dati specificati (solo unità canale) nell'area DM, a partire dal canale specificato.
WJ	AR AREA WRITE	Scrive i dati specificati (solo unità canale) nell'area ausiliaria, a partire dal canale specificato.
WE	EM AREA WRITE	Scrive i dati specificati (solo unità canale) nell'area EM, a partire dal canale specificato.
R#	SV READ 1	Legge la costante a 4 cifre in formato BCD o l'indirizzo del canale nel valore impostato relativo all'istruzione temporizzatore/contatore specificata.
R$	SV READ 2	Ricerca l'istruzione temporizzatore/contatore specificata a partire dall'indirizzo di programma specificato e legge la costante a 4 cifre o l'indirizzo del canale nel valore impostato.
R%	SV READ 3	Ricerca l'istruzione temporizzatore/contatore specificata a partire dall'indirizzo di programma specificato e legge la costante a 4 cifre in formato BCD o l'indirizzo del canale nel valore impostato.
W#	SV CHANGE 1	Modifica la costante a 4 cifre in formato BCD o l'indirizzo del canale nel valore impostato relativo all'istruzione temporizzatore/contatore specificata.
W$	SV CHANGE 2	Ricerca l'istruzione temporizzatore/contatore specificata a partire dall'indirizzo di programma specificato e modifica la costante a 4 cifre o l'indirizzo del canale nel valore impostato.
W%	SV CHANGE 3	Ricerca l'istruzione temporizzatore/contatore specificata a partire dall'indirizzo di programma specificato e modifica la costante a 4 cifre o l'indirizzo del canale nel valore impostato.
MS	STATUS READ	Legge lo stato operativo della CPU (modalità operativa, stato impostazione o ripristino forzato, stato di errore fatale).
SC	STATUS CHANGE	Modifica la modalità operativa della CPU.
MF	ERROR READ	Legge e cancella gli errori (fatali e non fatali) della CPU.
KS	FORCE SET	Imposta in modo forzato il bit specificato.
KR	FORCE RESET	Ripristina in modo forzato il bit specificato.
FK	MULTIPLE FORCE SET/RESET	Imposta o ripristina in modo forzato o annulla lo stato forzato dei bit specificati.
KC	FORCE SET/RESET CANCEL	Annulla lo stato forzato di tutti i bit impostati e ripristinati in modo forzato.
MM	PLC MODEL READ	Legge il tipo di modello del PLC.
TS	TEST	Restituisce, inalterato, un blocco di dati trasmesso dal computer host.
RP	PROGRAM READ	Legge il contenuto dell'area del programma utente della CPU in linguaggio macchina (codice oggetto).
WP	PROGRAM WRITE	Scrive il programma in linguaggio macchina (codice oggetto) trasmesso dal computer host nell'area del programma utente della CPU.
MI	I/O TABLE GENERATE	Crea una tabella di I/O registrati con la tabella di I/O effettiva.
QQMR	COMPOUND COMMAND	Registra i bit e i canali desiderati in una tabella.
QQIR	COMPOUND READ	Legge i canali e i bit registrati dalla memoria I/O.
XZ	ABORT (solo comando)	Interrompe il comando Host Link in corso di elaborazione.
**	INITIALIZE (solo comando)	Inizializza la procedura di controllo della trasmissione di tutti i PLC connessi al computer host.
IC	Comando non definito (solo risposta)	Questa risposta viene restituita se il codice di intestazione di un comando non viene riconosciuto.

Comandi FINS

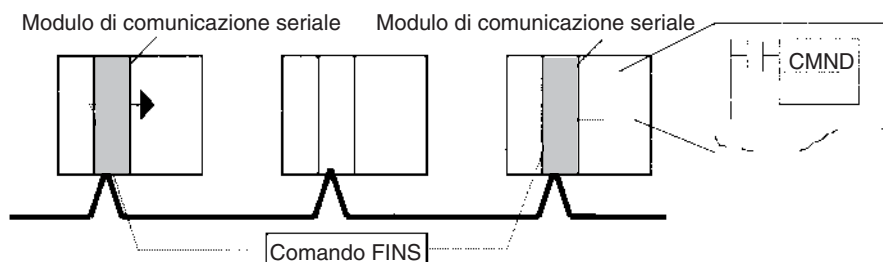
Nella tabella riportata di seguito sono elencati i comandi FINS. Per ulteriori informazioni sui comandi FINS, fare riferimento al *FINS Commands Reference Manual* (W227).

Tipo	Codice comando		Nome	Funzione
Accesso all'area di memoria I/O	01	01	MEMORY AREA READ	Legge dati consecutivi dall'area di memoria I/O.
	01	02	MEMORY AREA WRITE	Scrive dati consecutivi nell'area di memoria I/O.
	01	03	MEMORY AREA FILL	Inserisce i medesimi dati nell'intervallo della memoria I/O specificato.
	01	04	MULTIPLE MEMORY AREA READ	Legge dati non consecutivi dall'area di memoria I/O.
	01	05	MEMORY AREA TRANSFER	Copia e trasferisce dati consecutivi da un'area della memoria I/O a un'altra.
Accesso all'area dei parametri	02	01	PARAMETER AREA READ	Legge dati consecutivi dall'area dei parametri.
	02	02	PARAMETER AREA WRITE	Scrive dati consecutivi nell'area dei parametri.
	02	03	PARAMETER AREA FILL	Inserisce i medesimi dati nell'intervallo dell'area dei parametri specificato.
Accesso all'area del programma	03	06	PROGRAM AREA READ	Legge dati dall'area del programma utente.
	03	07	PROGRAM AREA WRITE	Scrive dati nell'area del programma utente.
	03	08	PROGRAM AREA CLEAR	Cancella l'intervallo specificato dell'area del programma utente.
Controllo dell'esecuzione	04	01	RUN	Cambia la modalità operativa della CPU passando alla modalità RUN, MONITOR o DEBUG.
	04	02	STOP	Cambia la modalità operativa della CPU passando alla modalità PROGRAM.
Lettura della configurazione	05	01	CONTROLLER DATA READ	Legge le informazioni della CPU.
	05	02	CONNECTION DATA READ	Legge i numeri di modello dei Moduli specificati.
Lettura dello stato	06	01	CONTROLLER STATUS READ	Legge le informazioni sullo stato della CPU.
	06	20	CYCLE TIME READ	Legge le informazioni relative ai tempi di ciclo medio, massimo e minimo.
Accesso all'orologio	07	01	CLOCK READ	Legge l'ora segnata dall'orologio.
	07	02	CLOCK WRITE	Imposta l'orologio.
Accesso ai messaggi	09	20	MESSAGE READ/CLEAR	Legge o cancella messaggi e messaggi FAL(S).
Diritto di accesso	0C	01	ACCESS RIGHT ACQUIRE	Acquisisce il diritto di accesso se non è detenuto da nessun altro dispositivo.
	0C	02	ACCESS RIGHT FORCED ACQUIRE	Acquisisce il diritto di accesso anche se è detenuto da un altro dispositivo.
	0C	03	ACCESS RIGHT RELEASE	Rilascia il diritto di accesso indipendentemente dal dispositivo che lo detiene.
Accesso agli errori	21	01	ERROR CLEAR	Cancella errori e messaggi di errore.
	21	02	ERROR LOG READ	Legge il log degli errori.
	21	03	ERROR LOG CLEAR	Azzera il puntatore del log degli errori.

Tipo	Codice comando		Nome	Funzione
Memoria file	22	01	FILE NAME READ	Legge le informazioni dei file della memoria file.
	22	02	SINGLE FILE READ	Legge la quantità di dati specificata dal punto specificato di un file.
	22	03	SINGLE FILE WRITE	Scrive la quantità di dati specificata dal punto specificato di un file.
	22	04	FILE MEMORY FORMAT	Formatta la memoria file.
	22	05	FILE DELETE	Elimina i file specificati dalla memoria file.
	22	07	FILE COPY	Copia un file all'interno della memoria file o tra due dispositivi di memoria file di un sistema.
	22	08	FILE NAME CHANGE	Modifica il nome di un file.
	22	0A	I/O MEMORY AREA FILE TRANSFER	Trasferisce o confronta i dati tra l'area della memoria I/O e la memoria file.
	22	0B	PARAMETER AREA FILE TRANSFER	Trasferisce o confronta i dati tra l'area dei parametri e la memoria file.
	22	0C	PROGRAM AREA FILE TRANSFER	Trasferisce o confronta i dati tra l'area del programma e la memoria file.
	22	15	CREATE/DELETE DIRECTORY	Crea o elimina una directory.
Stato forzato	23	01	FORCED SET/RESET	Imposta o ripristina in modo forzato o annulla lo stato forzato dei bit specificati.
	23	02	FORCED SET/RESET CANCEL	Annulla lo stato forzato di tutti i bit impostati e ripristinati in modo forzato.

Funzioni di comunicazione dei messaggi

I comandi FINS elencati nella tabella precedente possono essere trasmessi in rete da altri PLC alla CPU. Quando si trasmettono comandi FINS in rete, è bene tenere presente i seguenti punti:

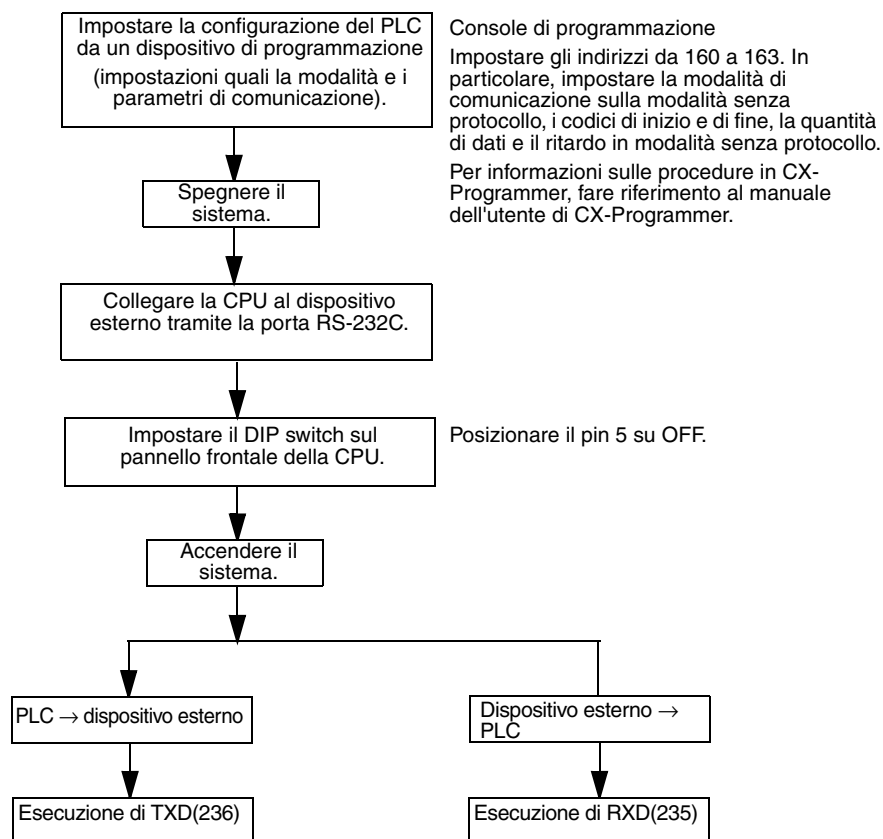
- I Moduli CPU bus, quali Moduli Controller Link o Ethernet, devono essere installati sul PLC locale e sul PLC di destinazione per la trasmissione dei comandi FINS.
- I comandi FINS vengono inviati tramite l'istruzione CMND(490) dal programma della CPU.
- I comandi FINS possono essere trasmessi attraverso un massimo di otto reti per le CPU della serie CS/CJ versione 2.0 o successiva e attraverso un massimo di tre reti per le altre CPU. Le reti possono essere dello stesso tipo o di tipo diverso.

Per ulteriori informazioni sulle funzioni di comunicazione dei messaggi, fare riferimento al manuale dell'operatore del Modulo CPU bus in uso.

6-3-2 Comunicazione senza protocollo

Nella tabella seguente sono riportate le funzioni di comunicazione senza protocollo disponibili nei PLC della serie CS/CJ.

Direzione del trasferimento	Metodo	Quantità massima di dati	Formato del frame		Altre funzioni
			Codice di inizio	Codice di fine	
Invio dati (PLC → dispositivo esterno)	Esecuzione di TXD(236) nel programma*	256 byte	Sì: da 00 a FF No: Nessuno	Sì: da 00 a FF o CR+LF No: Nessuno	Ritardo invio (ritardo tra esecuzione di TXD e invio dei dati dalla porta specificata): da 0 a 99.990 ms (unità: 10 ms)
Ricezione dati (Dispositivo esterno → PLC)	Esecuzione di RXD(235) nel programma	256 byte			---

Nota È possibile specificare un ritardo di invio o "ritardo in modalità senza protocollo" nella configurazione del PLC (indirizzo 162). Questa impostazione provoca un ritardo di 30 secondi al massimo tra l'esecuzione di TXD(236) e l'invio dei dati dalla porta specificata.

Procedura

Formati dei frame dei messaggi

I dati possono essere inseriti tra un codice di inizio e un codice di fine per la trasmissione con TXD(236) e i frame dello stesso formato possono essere ricevuti con RXD(235). Quando si trasmette con TXD(236), vengono inviati solo i dati dalla memoria I/O e, quando si riceve con RXD(235), nella memoria

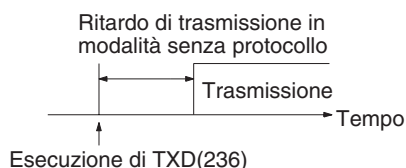
I/O vengono memorizzati solo tali dati. In modalità senza protocollo è possibile trasmettere fino a 256 byte, compresi i codici di inizio e di fine.

Nella tabella riportata di seguito sono elencati i formati di messaggio che è possibile impostare per trasmissioni e ricezioni in modalità senza protocollo. Il formato è determinato dall'impostazione dei codici di inizio (ST) e di fine (ED) nella configurazione del PLC.

Impostazione codice di inizio	Impostazione codice di fine		
	No	**Sì**	**CR+LF**
No	dati (*dati*: 256 byte max.)	*dati*+ED (*dati*: 255 byte max.)	*dati*+CR+LF (*dati*: 254 byte max.)
Sì	ST+*dati* (*dati*: 255 byte max.)	ST+*dati*+ED (*dati*: 254 byte max.)	ST+*dati*+CR+LF (*dati*: 253 byte max.)

- Quando si utilizza più di un codice di inizio, viene considerato valido il primo codice specificato.
- Quando si utilizza più di un codice di fine, viene considerato valido il primo codice specificato.

Nota

1. Se i dati in corso di trasferimento contengono un codice di fine, il trasferimento viene interrotto. In tal caso, è necessario modificare il codice di fine in CR+LF (ritorno a capo+avanzamento riga).
2. Nella configurazione del PLC è disponibile un'impostazione (indirizzo 162: ritardo in modalità senza protocollo) che consente di ritardare la trasmissione dei dati dopo l'esecuzione di TXD(236).

Per ulteriori informazioni su TXD(236) e RXD(235), fare riferimento al manuale *CJ-series Programmable Controllers Instructions Reference (W340)*.

6-3-3 NT Link (modalità 1:N)

I PLC della serie CS/CJ consentono la comunicazione con terminali programmabili mediante il protocollo NT Link in modalità 1:N.

Nota Non è possibile stabilire la comunicazione utilizzando il protocollo NT Link in modalità 1:1.

Oltre alle configurazioni NT Link standard precedenti, è possibile impostare configurazioni NT Link ad alta velocità utilizzando il menu di sistema del terminale programmabile e le impostazioni della configurazione del PLC riportate di seguito (non supportate dalle CPU CS1 della serie CS precedenti a EV1). Le configurazioni NT Link ad alta velocità sono tuttavia possibili solo con terminali programmabili NT31(C)-V2 o NT631(C)-V2.

Configurazione del PLC

Interfaccia di comunicazione	Indirizzo di impostazione dalla Console di programmazione	Nome	Contenuto delle impostazioni	Valori predefiniti	Altre condizioni
porta periferiche	144 Bit: da 8 a 11	Modalità di comunicazione seriale	02 esadecimale: NT Link (modalità 1:N)	00 esadecimale: Host Link	Posizionare su ON il pin 4 del DIP switch della CPU.
	145 Bit: da 0 a 7	Velocità di trasmissione	Da 00 a 09 esadecimale: NT Link standard 0A esadecimale: NT Link ad alta velocità (vedere nota 1)	00 esadecimale: NT Link standard	
	150 Bit: da 0 a 3	Numero massimo di unità in modalità NT Link	Da 0 a 7 esadecimale	0 esadecimale (numero max. unità = 0)	---
porta RS-232C	160 Bit: da 8 a 11	Modalità di comunicazione seriale	02 esadecimale: NT Link (modalità 1:N)	00 esadecimale: Host Link	Posizionare su OFF il pin 5 del DIP switch della CPU.
	161 Bit: da 0 a 7	Velocità di trasmissione	Da 00 a 09 esadecimale: NT Link standard 0A esadecimale: NT Link ad alta velocità (vedere nota 1)	00 esadecimale: NT Link standard	
	166 Bit: da 0 a 3	Numero massimo di unità in modalità NT Link	Da 0 a 7 esadecimale	0 esadecimale (numero max. unità = 0)	---

Nota Durante l'impostazione con CX-Programmer, impostare la velocità di trasmissione su 115.200 bps.

Menu di sistema del terminale programmabile

Impostare il terminale programmabile come descritto di seguito.

1,2,3...

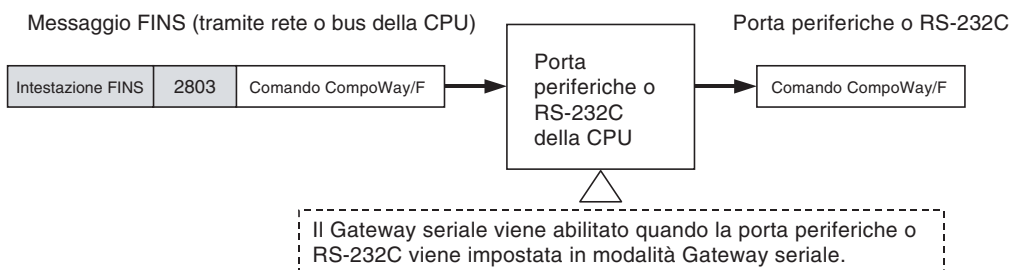
1. Selezionare NT Link (1:N) da Comm. A Method (Metodo com. A) o Comm. B Method (Metodo com. B) nel sottomenu Memory Switch (Selezione memoria) del menu System (Sistema) del terminale programmabile.
2. Toccare il comando SET per impostare Comm. Speed (Velocità com.) su High Speed (Alta velocità).

6-3-4 Gateway seriale della CPU

Introduzione al Gateway seriale

I messaggi (comandi) FINS ricevuti vengono automaticamente convertiti nel protocollo specificato e quindi inviati tramite comunicazione seriale. Anche le risposte vengono automaticamente convertite. Quando si utilizza la porta periferiche o la porta RS-232C della CPU, i messaggi FINS possono essere convertiti nel seguente protocollo:

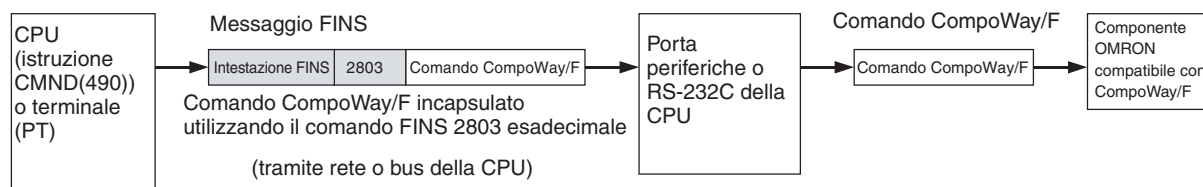
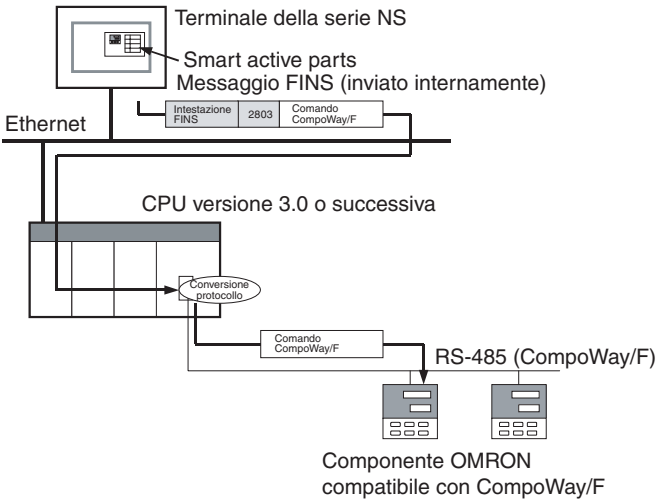
- CompoWay/F

Il Gateway seriale viene abilitato quando la modalità di comunicazione seriale viene impostata su Gateway seriale.

Caratteristiche del Gateway seriale

Specifica	Dettagli
Origine di conversione	Comando FINS (ricevuto tramite rete FINS, FINS Host Link, bus di periferica, NT Link o bus della CPU)
Funzione di conversione	I comandi FINS ricevuti vengono convertiti in base ai seguenti valori prima dell'invio alla porta seriale (porta periferiche o RS-232C) della CPU. 2803 esadecimale: l'intestazione FINS viene rimossa e il comando FINS viene convertito in comando CompoWay/F
Dopo la conversione	Comandi CompoWay/F
Metodo di comunicazione seriale	Comunicazioni half-duplex 1:N
Numero massimo di Moduli connessi	31 slave
Modalità di comunicazione seriale supportate	Modalità del Gateway seriale
Monitoraggio del timeout di risposta	Viene monitorato il tempo dall'invio di un messaggio convertito nel protocollo CompoWay/F utilizzando il Gateway seriale alla ricezione di una risposta (abilitata in modalità Gateway seriale o protocol macro). Valore predefinito: 5 s, gamma di impostazione: 0,1 ... 25,5 s **Nota** In caso di timeout all'origine del comando FINS viene restituito il codice di fine FINS (0205 esadecimale: timeout di risposta).
Ritardo di invio	Nessuno

Conversione da FINS in CompoWay/F

È possibile accedere a componenti OMRON collegati serialmente alla porta periferiche o RS-232C della CPU utilizzando CompoWay/F da un PLC o da un terminale specificando comandi CompoWay/F all'interno di messaggi FINS.

- Messaggio FINS inviato: intestazione FINS + codice di comando FINS 2803 esadecimale + comando CompoWay/F
- Messaggio dopo la conversione: comando CompoWay/F

Per informazioni dettagliate sul codice di comando FINS 2803 esadecimale, fare riferimento al manuale *Communications Commands Reference Manual* (W342).

Componenti compatibili come slave con CompoWay/F

Componente		Serie del modello
Termoregolatori	Thermac NEO	E5GN (componenti G)
		E5CN
		E5EN
		E5AN
	Thermac R	E5AR
		E5ER
	Termoregolatori con attacco undecal	E5ZN
	Schede di regolatori digitali	E5ZM
	Regolatori digitali	ES100X
Temporizzatori/ Contatori	Temporizzatori/Contatori	H8GN (componenti G)
Strumenti di misura digitali	Strumenti di misura digitali	K3GN (componenti G) K3NX
	Misuratori di celle di carico digitali	K3NV
	Misuratori di impulsi/rotazioni digitali	K3NR
	Strumenti di misura di incremento digitali	K3NP
	Misuratori di intervalli di tempo digitali	K3NC
	Misuratori di processo/temperatura digitali	K3NH
Sensori di spostamento	Moduli di interfaccia di comunicazione ZX	ZX-SF11
Posizionatori angolari	---	3F88L-160, 3F88L-162
Controllori di sicurezza	---	F3SX

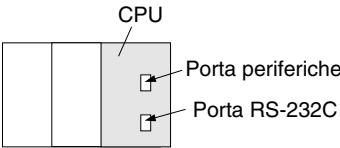
Esempi di configurazione del sistema

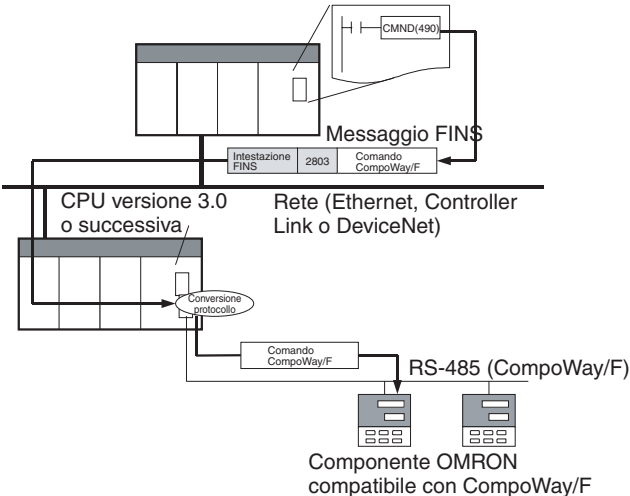
Esecuzione di smart active parts utilizzando un terminale della serie NS (inviando messaggi FINS interni)

In questo caso la CPU converte i messaggi FINS nel protocollo CompoWay/F per l'invio.

Accesso da un terminale tramite Ethernet o NT Link seriale	Dettagli	Utilizzo di tabelle di routing per trattare il percorso di comunicazione seriale come una rete
Terminale della serie NS Smart active parts Messaggio FINS (inviato internamente) Intestazione FINS · 2803 · Comando CompoWay/F Ethernet CPU versione 3.0 o successiva Conversione protocollo Comando CompoWay/F RS-485 (CompoWay/F) Componente OMRON compatibile con CompoWay/F **Nota** Se il terminale della serie NS è collegato serialmente al PLC, si utilizza la modalità di comunicazione seriale (NT Link 1:N) e il terminale della serie NS invia comandi FINS incapsulati all'interno di comandi NT Link utilizzando componenti attivi intelligenti, la CPU rimuove l'intestazione NT Link e altre sezioni dal comando ricevuto, convertendolo in un comando FINS, e trasferisce tale comando alla porta seriale della CPU. La CPU utilizza il Gateway seriale per convertire il comando nel protocollo specificato. Questa operazione consente ai dispositivi collegati serialmente di accedere alla porta seriale della CPU da componenti attivi intelligenti tramite un terminale della serie NS.	È possibile accedere tramite comunicazione seriale utilizzando CompoWay/F da un terminale collegato alla rete eseguendo un componente attivo intelligente connesso serialmente che invia automaticamente un comando FINS interno.	Opzionale

Nota 1. L'intestazione FINS contiene le seguenti informazioni:

- Indirizzo della rete di destinazione remota (DNA)
  - Utilizzando tabelle di routing che trattano il percorso di comunicazione seriale come una rete. In questo caso nelle tabelle di routing l'indirizzo di rete corrisponde a una porta seriale.
  - Senza utilizzare tabelle di routing che trattano il percorso di comunicazione seriale come una rete. In questo caso l'indirizzo della rete indica il PLC remoto.
- Indirizzo del nodo di destinazione remoto (DA1)
  - Utilizzando tabelle di routing che trattano il percorso di comunicazione seriale come una rete. In questo caso il valore è 00 esadecimale (comunicazione interna del PLC locale).
  - Senza utilizzare tabelle di routing che trattano il percorso di comunicazione seriale come una rete. In questo caso l'indirizzo del nodo indica il PLC remoto.
- Indirizzo del modulo di destinazione remoto (DA2)
  Indirizzo del modulo della porta seriale

Porta seriale della CPU	Indirizzo del modulo della porta seriale
Porta periferiche	FD esadecimale (253 decimale)
Porta RS-232C	FC esadecimale (252 decimale)

2. Il contenuto del comando CompoWay/F all'interno del messaggio FINS inviato è:
   Numero nodo + sottoindirizzo + SID + testo del comando (in formato ASCII)
   STX, ETX+BCC non sono richiesti quando si invia un messaggio FINS. Vengono aggiunti automaticamente per la comunicazione seriale.

Invio di messaggi FINS utilizzando CMND(490) in un programma ladder della CPU

In questo caso la CPU converte i messaggi FINS nel protocollo CompoWay/F per l'invio.

Accesso dalla CPU (sullo stesso PLC)

Conversione seriale	Dettagli	Utilizzo di tabelle di routing per trattare il percorso di comunicazione seriale come una rete
CPU versione 3.0 o successiva Porta RS-232C CMND(490) Messaggio FINS Intestazione FINS | 2803 | Comando CompoWay/F Conversione protocollo Comando CompoWay/F RS-485 (CompoWay/F) Componente OMRON compatibile con CompoWay/F	È possibile accedere a componenti OMRON collegati serialmente alla porta RS-232C della CPU utilizzando CompoWay/F da una CPU sullo stesso PLC.	Opzionale

Accesso dalla CPU (PLC sulla rete)

Da messaggio FINS a seriale	Dettagli	Utilizzo di tabelle di routing per trattare il percorso di comunicazione seriale come una rete
	È possibile accedere a componenti OMRON collegati serialmente alla porta RS-232C della CPU utilizzando CompoWay/F da una CPU su un PLC connesso in rete.	Opzionale

Frame di comunicazione

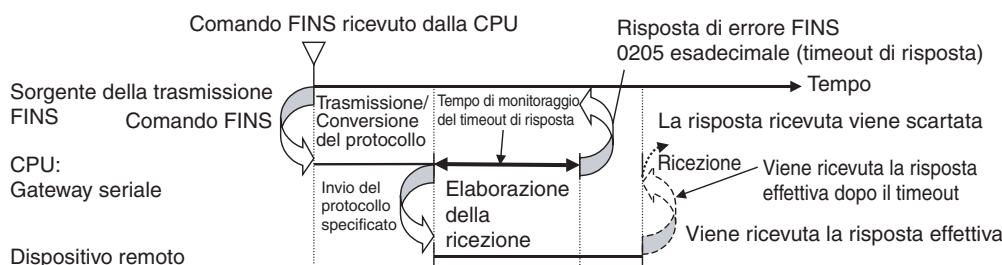
Frame di comando

Nota Poiché i comandi CompoWay/F utilizzano ASCII come codice di trasmissione, accertarsi di utilizzare tale codice per il comando CompoWay/dopo il codice di comando FINS 2803 esadecimale (dal numero del nodo al testo) utilizzando CMND(490) o un'altra istruzione.
Esempio: se il comando CompoWay/F MRC SRC è “01” “02” (le virgolette racchiudono caratteri ASCII), 0, 1, 0 e 2 devono essere interpretati come caratteri ASCII. Impostare quindi “01” come 3031 esadecimale (non 01 esadecimale) e “02” come 3032 esadecimale (non 02 esadecimale).

Inoltre, per scrivere il comando CompoWay/F nell'area di memorizzazione dei comandi utilizzando l'istruzione CMND(490) nell'ordine dei frame (senza creare byte vuoti), il componente SID del comando CompoWay/F richiede 1 byte ASCII contenente il valore 30 esadecimale, in modo che i componenti successivi (s+3 e oltre) debbano essere impostati in un byte ciascuno.

Monitoraggio del timeout di risposta (modalità Gateway seriale)

In modalità Gateway seriale viene monitorato il tempo che trascorre dall'invio del messaggio convertito nel protocollo specificato dal Gateway seriale alla ricezione della risposta da parte del dispositivo remoto. La gamma di impostazione per un valore specificato dall'utente è 0,1 … 25,5 s e l'impostazione predefinita è 5 s.

Se sulla porta seriale non viene ricevuta alcuna risposta entro il tempo specificato, all'origine del comando FINS viene restituita una risposta di errore FINS (codice di fine: 0205 esadecimale - timeout di risposta). Tuttavia, se viene ricevuta una risposta dopo il timeout, la risposta ricevuta per ciascun protocollo viene scartata e all'origine del comando FINS non viene restituita alcuna risposta FINS.

6-3-5 Collegamenti seriali tra PLC (solo CPU CJ1M)

Informazioni generali

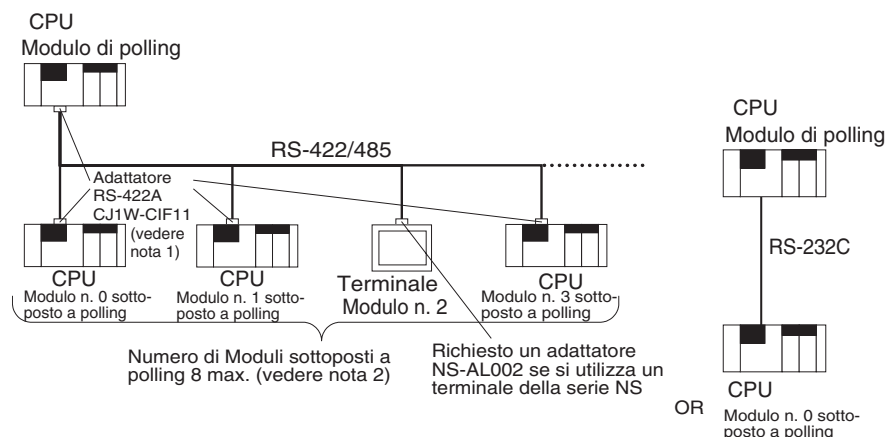
I collegamenti seriali tra PLC sono supportati solo dalle CPU CJ1M e consentono di scambiare dati tra CPU CJ1M mediante le porte RS-232C integrata senza richiedere interventi di programmazione speciale. I canali vengono assegnati in memoria nei canali per il collegamento seriale tra PLC (da CIO 3100 a CIO 3199). Le connessioni RS-232C vengono utilizzate per la comunicazione tra CPU. In alternativa, è possibile utilizzare connessioni RS-422A/485 collegando adattatori da RS-232C a RS-422A/485 alle porte RS-232C.

Per la conversione da RS-232C a RS-422A/485 è possibile utilizzare gli adattatori RS-422A CJ1W-CIF11.

È inoltre possibile utilizzare un terminale programmabile impostato per la comunicazione NT Link (1:N) sulla stessa rete. Il terminale programmabile sottoposto a polling utilizza la rete per comunicare con la CPU che esegue il polling attraverso una connessione NT Link in modalità 1:N. Tuttavia, quando si collega un terminale programmabile, gli indirizzi nei canali per il collegamento seriale tra PLC corrispondenti al numero di modulo del terminale non sono definiti.

Specifiche

Elemento	Specifiche
Metodo di collegamento	Connessione RS-232C o RS-422A/485 tramite porta RS-232C della CPU.
Area dati assegnata	Canali per il collegamento seriale tra PLC: da CIO 3100 a CIO 3199 (è possibile assegnare fino a 10 canali per ciascuna CPU)
Numero di Moduli	Massimo 9 Moduli, compresi un Modulo di polling e otto Moduli sottoposti a polling (è possibile inserire sulla stessa rete un terminale programmabile configurato per NT Link (1:N), ma in tal caso verrà considerato come uno degli 8 Moduli sottoposti a polling)

Configurazione del sistema

Nota

1. L'adattatore CJ1W-CIF11 non dispone di isolamento e la distanza di trasmissione totale massima è di 50 metri. Se la distanza di trasmissione totale è maggiore di 50 metri, utilizzare l'adattatore NT-AL001 isolato anziché l'adattatore CJ1W-CIF11. Se si utilizza solo l'adattatore NT-AL001, la distanza di trasmissione totale massima sarà pari a 500 metri.
2. Se nella rete viene inserito un terminale programmabile impostato per la comunicazione seriale tra PLC, è possibile collegare al Modulo di polling 8 Moduli al massimo, e il terminale deve essere considerato come uno dei Moduli sottoposti a polling.

Metodi di aggiornamento dei dati

Per aggiornare i dati, è possibile utilizzare i seguenti metodi:

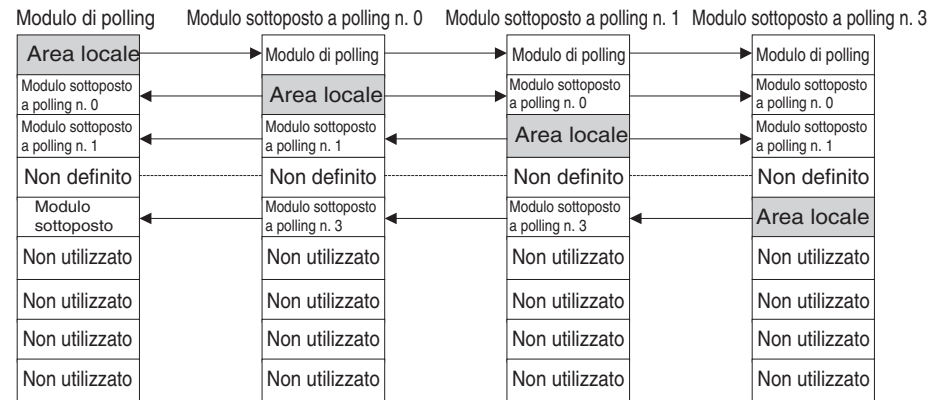
- Metodo di collegamento globale
- metodo di collegamento tramite Modulo di polling

Metodo di collegamento globale

I dati di tutti i nodi che partecipano ai collegamenti seriali tra PLC vengono riflessi sia nel Modulo di polling che nei Moduli sottoposti a polling. Le uniche

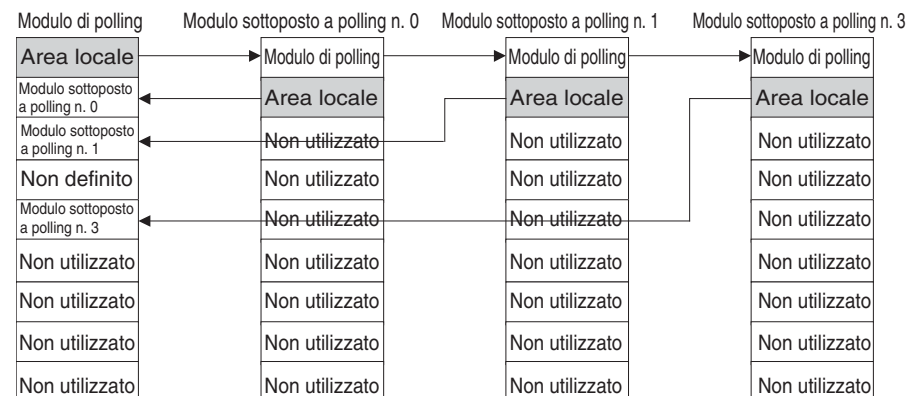
eccezioni riguardano l'indirizzo assegnato al numero di modulo dell'eventuale terminale programmabile collegato e gli indirizzi dei Moduli sottoposti a polling non attivi in rete. Queste aree di dati risultano non definite su tutti i nodi.

Esempio metodo di collegamento globale (numero di modulo più alto: 3.

Nello schema riportato di seguito il Modulo n. 2 sottoposto a polling è un terminale programmabile o un Modulo non attivo in rete; pertanto, l'area assegnata al Modulo n. 2 risulta non definita su tutti i nodi.

Metodo di collegamento tramite Modulo di polling

I dati di tutti i Moduli sottoposti a polling che partecipano ai collegamenti seriali tra PLC vengono riflessi solo nel Modulo di polling, mentre ciascun Modulo sottoposto a polling riflette solo i dati del Modulo di polling. Il vantaggio offerto da questo metodo risiede nel fatto che l'indirizzo assegnato per i dati del Modulo locale sottoposto a polling è lo stesso per tutti i Moduli, ed è pertanto possibile accedere ai dati tramite la normale programmazione ladder. Le aree assegnate per i numeri di modulo del terminale programmabile o dei Moduli sottoposti a polling non attivi in rete risultano non definite solo sul Modulo di polling.

Esempio metodo di collegamento tramite Modulo di polling (numero di modulo più alto: 3.

Nello schema riportato di seguito il Modulo n. 2 sottoposto a polling è un terminale programmabile o un Modulo non attivo in rete; pertanto, la corrispondente area assegnata risulta non definita sul Modulo di polling.

Canali assegnati

Metodo di collegamento globale

Indirizzo

CIO 3100

Canali per il collegamento seriale tra PLC

CIO 3199

Canali di collegamento	1 canale	2 canali	3 canali	a	10 canali
Modulo di polling	CIO 3100	Da CIO 3100 a CIO 3109	Da CIO 3100 a CIO 3109		Da CIO 3100 a CIO 3109
Modulo n. 0 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 1 sottoposto a polling	CIO 3102	Da CIO 3103 a CIO 3104	Da CIO 3106 a CIO 3108		Da CIO 3120 a CIO 3129
Modulo n. 2 sottoposto a polling	CIO 3103	Da CIO 3106 a CIO 3107	Da CIO 3109 a CIO 3111		Da CIO 3130 a CIO 3139
Modulo n. 3 sottoposto a polling	CIO 3104	Da CIO 3100 a CIO 3108	Da CIO 3112 a CIO 3113		Da CIO 3140 a CIO 3149
Modulo n. 4 sottoposto a polling	CIO 3105	Da CIO 3110 a CIO 3119	Da CIO 3116 a CIO 3115		Da CIO 3150 a CIO 3159
Modulo n. 5 sottoposto a polling	CIO 3106	Da CIO 3112 a CIO 3113	Da CIO 3118 a CIO 3120		Da CIO 3160 a CIO 3169
Modulo n. 6 sottoposto a polling	CIO 3107	Da CIO 3114 a CIO 3115	Da CIO 3121 a CIO 3123		Da CIO 3170 a CIO 3179
Modulo n. 7 sottoposto a polling	CIO 3108	Da CIO 3116 a CIO 3117	Da CIO 3124 a CIO 3126		Da CIO 3180 a CIO 3189
non utilizzato	Da CIO 3109 a CIO 3199	Da CIO 3118 a CIO 3199	Da CIO 3127 a CIO 3199		Da CIO 3190 a CIO 3199

Metodo di collegamento tramite Modulo di polling

Indirizzo

CIO 3100

Canali per il collegamento seriale tra PLC

CIO 3199

Canali di collegamento	1 canale	2 canali	3 canali	a	10 canali
Modulo di polling	CIO 3100	Da CIO 3100 a CIO 3109	Da CIO 3100 a CIO 3109		Da CIO 3100 a CIO 3109
Modulo n. 0 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 1 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 2 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 3 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 4 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 5 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 6 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
Modulo n. 7 sottoposto a polling	CIO 3101	Da CIO 3102 a CIO 3103	Da CIO 3103 a CIO 3105		Da CIO 3110 a CIO 3119
non utilizzato	Da CIO 3102 a CIO 3199	Da CIO 3104 a CIO 3199	Da CIO 3106 a CIO 3199		Da CIO 3120 a CIO 3199

Procedura

I collegamenti seriali tra PLC funzionano in base alle seguenti impostazioni della configurazione del PLC.

Impostazioni per il Modulo di polling

1,2,3...
1. Impostare la modalità di comunicazione seriale della porta di comunicazione RS-232C su Serial PLC Links (Polling Unit) [Collegamenti seriali tra PLC (Modulo di polling)].
2. Impostare il metodo di collegamento su Complete Link Method (Metodo di collegamento globale) o Polling Unit Link Method (metodo di collegamento tramite Modulo di polling).
3. Impostare il numero di canali di collegamento (massimo 10 canali per Modulo).
4. Impostare il numero massimo di moduli per i collegamenti seriali tra PLC (tra 0 e 7).

Impostazioni per i Moduli sottoposti a polling

1,2,3...
1. Impostare la modalità di comunicazione seriale della porta di comunicazione RS-232C su Serial PLC Links (Polled Unit) [Collegamenti seriali tra PLC (Modulo sottoposto a polling)].
2. Impostare il numero di modulo del Modulo sottoposto a polling che partecipa al collegamento seriale tra PLC.

Configurazione del PLC

Impostazioni per il Modulo di polling

Elemento		Indirizzo del PLC		Valore impostato	Valore predefinito	Frequenza di aggiornamento
		Canale	Bit			
Impostazione della porta RS-232C	Modalità di comunicazione seriale	160	Da 11 a 08	8 esadecimale: Modulo di polling dei collegamenti seriali tra PLC	0 esadecimale	Ogni ciclo
	Velocità di trasmissione della porta	161	Da 07 a 00	Da 00 a 09 esadecimale: standard 0A esadecimale: alta velocità (vedere nota 2)	00 esadecimale	
	Metodo di collegamento	166	15	0: collegamento globale 1: collegamento tramite Modulo di polling	0	
	Numero di canali di collegamento		Da 07 a 04	Da 1 ad A esadecimale	0 esadecimale (vedere nota 1)	
	Numero di modulo più alto		Da 03 a 00	Da 0 a 7 esadecimale	0 esadecimale	

Nota
1. Quando si utilizza l'impostazione predefinita (0 esadecimale), vengono automaticamente assegnati 10 canali (A esadecimale).
2. Quando si imposta CX-Programmer, specificare 115.200 bit/s.

Impostazioni per il Modulo sottoposto a polling

Elemento		Indirizzo del PLC		Valore impostato	Valore predefinito	Frequenza di aggiornamento
		Canale	Bit			
Impostazioni della porta RS-232C	Modalità di comunicazione seriale	160	Da 11 a 08	7 esadecimale: Modulo sottoposto a polling del collegamento seriale tra PLC	0 esadecimale	Ogni ciclo
	Velocità di trasmissione della porta	161	Da 07 a 00	Da 00 a 09 esadecimale: standard 0A esadecimale: alta velocità (vedere nota)	00 esadecimale	
	Numero di modulo del Modulo sottoposto a polling	167	Da 03 a 00	Da 0 a 7 esadecimale	0 esadecimale	

Nota Quando si imposta CX-Programmer, specificare 115.200 bit/s.

Flag dell'area ausiliaria correlati

Nome	Indirizzo	Dettagli	Lettura/ Scrittura	Frequenza di aggiornamento
Flag di errore di comunicazione sulla porta RS-232C	A39204	Viene attivato quando si verifica un errore di comunicazione sulla porta RS-232C. 1: errore 0: normale	Lettura	• Viene cancellato all'accensione. • Viene attivato quando si verifica un errore di comunicazione sulla porta RS-232C. • Viene disattivato quando la porta viene riavviata. • È disabilitato in modalità bus di periferica e in modalità NT Link.
Flag di comunicazione in corso tra PT e porta RS-232C (vedere nota)	Da A39300 ad A39307	Quando la porta RS-232C è in uso in modalità NT Link, il bit corrispondente al Modulo che esegue la comunicazione viene attivato. I bit da 00 a 07 corrispondono rispettivamente ai numeri di modulo da 0 a 7. 1: comunicazione in corso 0: nessuna comunicazione	Lettura	• Viene cancellato all'accensione. • Attiva il bit corrispondente al numero di modulo del Modulo o del terminale programmabile sottoposto a polling che comunica tramite la porta RS-232C in modalità NT Link o in modalità di collegamento seriale tra PLC. • I bit da 00 a 07 corrispondono rispettivamente ai numeri di modulo da 0 a 7.
Bit di riavvio della porta RS-232C	A52600	Impostare questo bit su ON per riavviare la porta RS-232C.	Lettura/ Scrittura	• Viene cancellato all'accensione. • Viene attivato al riavvio della porta RS-232C (tranne durante la comunicazione in modalità bus di periferica). Nota: a seconda del sistema, è possibile che il bit venga automaticamente disattivato al termine del processo di riavvio.
Flag di errore della porta RS-232C	Da A52800 ad A52807	Quando si verifica un errore sulla porta RS-232C, viene memorizzato il codice di errore corrispondente. Bit 00: non utilizzato Bit 01: non utilizzato Bit 02: errore di parità Bit 03: errore di frame Bit 04: errore di overrun Bit 05: errore di timeout Bit 06: non utilizzato Bit 07: non utilizzato	Lettura/ Scrittura	• Viene cancellato all'accensione. • Quando si verifica un errore sulla porta RS-232C, viene memorizzato il codice di errore corrispondente. • A seconda del sistema, è possibile che il flag venga cancellato al riavvio della porta RS-232C. • È disabilitato in modalità bus di periferica. • In modalità NT Link, è abilitato solo il bit 05 (errore di timeout). In modalità di collegamento seriale tra PLC sono abilitati solo i bit riportati di seguito. Errore del Modulo di polling: Bit 05: errore di timeout • Errore di CHECK del Modulo sottoposto a polling Bit 05: errore di timeout Bit 04: errore di overrun Bit 03: errore di frame
Flag di modifica delle impostazioni della porta RS-232C	A61902	Viene attivato quando è in corso la modifica delle condizioni di comunicazione della porta RS-232C. 1: modificate 0: invariate	Lettura/ Scrittura	• Viene cancellato all'accensione. • Viene attivato quando vengono modificate le impostazioni relative alle condizioni di comunicazione della porta RS-232C. • Viene attivato quando viene eseguita l'istruzione CHANGE SERIAL PORT SETUP (STUP(237)). • Viene nuovamente disattivato al termine della modifica delle impostazioni.

Nota Analogamente alle connessioni NT Link (1:N) esistenti, è possibile verificare lo stato di comunicazione in corso/nessuna comunicazione dei terminali programmabili che partecipano al collegamento seriale tra PLC dal Modulo di polling (CPU) leggendo il flag di comunicazione in corso tra PT e porta RS-232C (bit da 00 a 07 di A393 per i numeri di modulo da 0 a 7).

6-4 Modifica della modalità di aggiornamento dei valori attuali di temporizzatori e contatori

6-4-1 Informazioni generali

In precedenza, le CPU CS1 utilizzavano solo il formato BCD come modalità di aggiornamento dei valori attuali di temporizzatori e contatori. Di conseguenza, tutte le impostazioni di temporizzatore e contatore venivano immesse come valori in formato BCD. Altre CPU (vedere note 1 e 2) consentono di utilizzare sia la modalità BCD che la modalità binaria per l'aggiornamento dei valori attuali delle istruzioni di temporizzatore e contatore (vedere nota 3).

Quando si utilizza la modalità binaria, la gamma di impostazione del temporizzatore/contatore, precedentemente stabilita tra 0 e 9.999, può essere espansa da 0 a 65.535. Per i valori impostati del temporizzatore/contatore è anche possibile utilizzare dati binari calcolati con altre istruzioni. È possibile specificare la modalità di aggiornamento dei valori attuali di temporizzatori e contatori quando il valore impostato per il temporizzatore o il contatore viene specificato come indirizzo (riferimento indiretto). L'impostazione della modalità BCD o binaria determina l'interpretazione del contenuto del canale indirizzato come valore BCD o binario.

Per quanto riguarda gli operandi dell'istruzione, esistono alcune differenze tra modalità BCD e modalità binaria. Pertanto, prima di modificare la modalità di aggiornamento dei valori attuali di temporizzatori e contatori, è necessario verificare e comprendere a fondo le differenze tra queste modalità.

Nota

1. Di seguito sono riportate le CPU, diverse dalle CPU CS,1 che supportano entrambe le modalità:
   - CPU CS1-H
   - CPU CJ1-H
   - CPU CJ1M
   - CPU CS1D
2. Per le CPU CS1-H, CJ1-H, CJ1M o CS1D prodotte fino al 31 maggio 2002 in cui la modalità di aggiornamento dei valori attuali di temporizzatori e contatori viene impostata come modalità binaria, durante il monitoraggio del codice mnemonico dalla Console di programmazione, il codice della modalità binaria viene visualizzato come codice mnemonico o istruzione BCD (ad esempio, TIMX #0000 &16 viene visualizzato come TIM #0000 &16) ma le operazioni vengono eseguite in modalità binaria.
3. È possibile selezionare la modalità di aggiornamento dei valori attuali solo con CX-Programmer versione 3.0. La selezione della modalità non è supportata da CX-Programmer 2.1 o versione precedente né dalle Console di programmazione.
4. CX-Programmer versione 2.1 o precedente non consente di leggere programmi utente per la CPU contenenti istruzioni in modalità binaria, ma consente di leggere quelli impostati utilizzando istruzioni in modalità BCD.

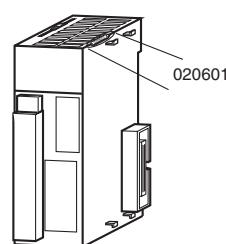
6-4-2 Specifiche funzionali

Elemento	Dettagli		
Metodo di impostazione della modalità di aggiornamento dei valori attuali di temporizzatori/ contatori	Deve essere impostata mediante CX-Programmer versione 3.0 (non supportata da CX-Programmer 2.1 o versione precedente). Viene impostata nelle proprietà del PLC di CX-Programmer versione.3.0.		
CPU supportate	CPU CS1-H/CJ1-H, n. lotto 020601 (data di produzione: 1 giugno 2002) e successive (vedere nota 1) e CPU CJ1M e CS1D.		
Modalità	**Modalità BCD**	**Modalità binaria**	
Codice mnemonico	Come per i modelli precedenti Esempio: TIM	Viene aggiunta una X al codice mnemonico della modalità BDC Esempio: TIMX	
Codice funzione	Come per i modelli precedenti	Nuovi codici	
Gamma valore attuale/valore impostato	Da #0000 a #9999	Da &0 a &65536	Da #0000 a #FFFF
Visualizzazione del valore attuale sul dispositivo di programmazione (CX-Programmer versione 3.0 o Console di programmazione)	Formato BCD Esempio: #0100	Formato decimale Esempio: &100	Formato esadecimale Esempio: #64

Nota Per le CPU CS1-H/CJ1-H prodotte fino al 31 maggio 2002 in cui la modalità di aggiornamento dei valori attuali di temporizzatori e contatori viene impostata come modalità binaria, durante il monitoraggio del codice mnemonico dalla Console di programmazione, il codice della modalità binaria viene visualizzato come codice mnemonico o istruzione BCD (ad esempio, TIMX #0000 &16 viene visualizzato come TIM #0000 &16) ma le operazioni vengono eseguite in modalità binaria.

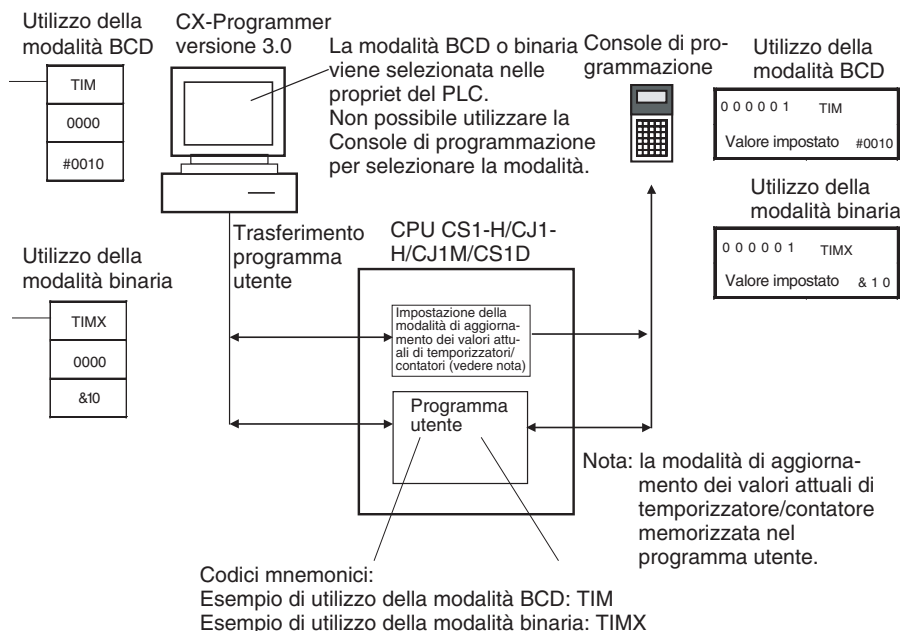
Verifica del numero di lotto della CPU

1,2,3...

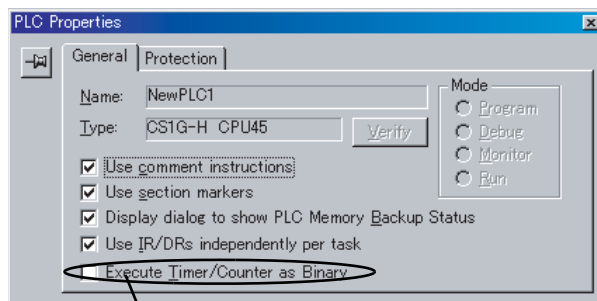
1. Il numero di lotto è stampato sulla parte inferiore del pannello frontale (serie CS) o nell'angolo a destra della parte superiore del Modulo (serie CJ) ed è costituito dalle ultime due cifre dell'anno, dal mese e dal giorno, in quest'ordine, come illustrato nello schema seguente.
   Ad esempio, 020601 (data di produzione: 1 giugno 2002)

2. Verificare la modalità selezionata. A tale scopo, portare CX-Programmer in linea, aprire la finestra della tabella di I/O e selezionare ***Unit Information*** - ***CPU Unit*** (Informazioni sul modulo - CPU). Il numero di lotto verrà visualizzato nel formato illustrato nella schema precedente, vale a dire le ultime due cifre dell'anno, il mese e il giorno, in quest'ordine.

6-4-3 Selezione e conferma della modalità BCD o binaria

Quando si crea un nuovo programma, in CX-Programmer versione 3.0 è possibile selezionare la modalità BCD o la modalità binaria nelle impostazioni delle proprietà del PLC.

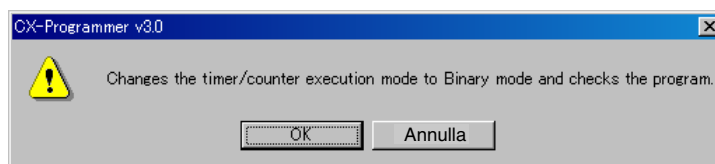
Nota La selezione della modalità BCD o binaria è supportata solo da CX-Programmer versione 3.0 o successiva. Con CX-Programmer 2.1 o versione precedente non è possibile effettuare questa selezione.

Selezione della modalità BCD o binaria

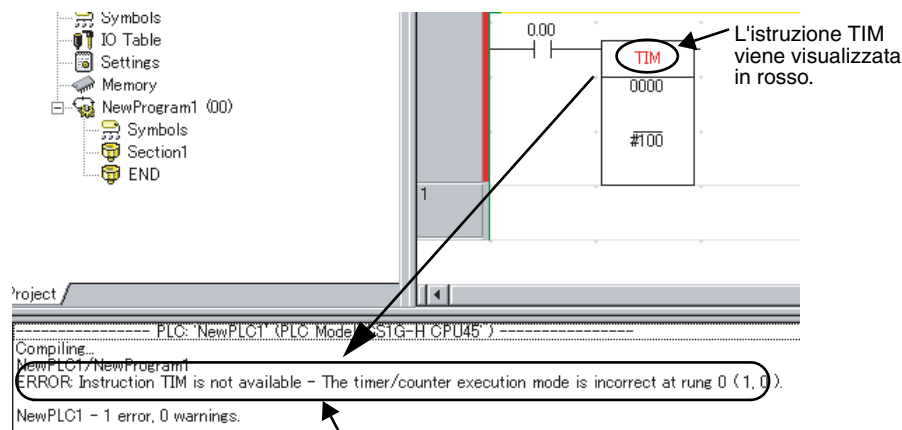
1,2,3...

1. Selezionare il nome del PLC, fare clic con il pulsante destro del mouse e scegliere ***PLC Properties*** (Proprietà PLC).

Selezionare questa casella di controllo per abilitare l'impostazioı

2. Fare clic sulla scheda **General** (Generale) e selezionare ***Execute Timers/ Counters as Binary*** (Esegui temporizzatori/contatori come binari).
   - Non selezionata (impostazione predefinita): Modalità BCD
   - Selezionata: Modalità binaria

   Il valore di impostazione della modalità di aggiornamento dei valori attuali di temporizzatore/contatore impostato nelle proprietà del PLC verrà memorizzato nella memoria utente della CPU quando il programma utente verrà trasferito da CX-Programmer alla CPU.

Quando si modifica questa impostazione, viene visualizzata la seguente finestra di dialogo.

Fare clic su **OK** per eseguire la verifica del programma. I risultati della verifica verranno visualizzati nella finestra di output.

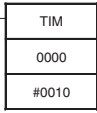
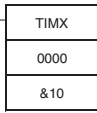
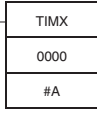
Esempio è stata utilizzata l'istruzione TIM anche se la modalità è stata modificata in binaria.

I risultati della verifica del programma vengono visualizzati nella finestra di output. Esempio la modalità operativa del temporizzatore/contatore è diversa, pertanto non è possibile utilizzare l'istruzione TIM.

Conferma della modalità BCD o binaria

È possibile utilizzare il flag dell'area ausiliaria relativo alla modalità di aggiornamento dei valori attuali di temporizzatore/contatore (A09915) per verificare se una CPU sta operando in modalità BCD o binaria.

Nome	Indirizzo	Dettagli
Flag della modalità di aggiornamento dei valori attuali di temporizzatore/contatore	A09915	0: modalità BCD 1: Modalità binaria

6-4-4 Codici mnemonici e dati in modalità BCD o binaria

Codici mnemonici in modalità BCD o binaria

In modalità binaria, i codici mnemonici sono riconoscibili dall'aggiunta del suffisso X al codice mnemonico della modalità BCD.

Esempio codici mnemonici per l'istruzione TIMER

Modalità BCD: TIM

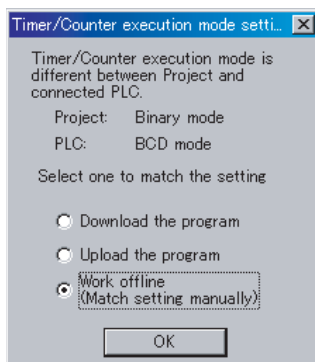
Modalità binaria: TIMX

Visualizzazione dei dati in modalità BCD o binaria

Proprietà del PLC	Significato dei simboli di ingresso/ visualizzazione	Gamma di impostazione	Esempio numero temporizzatore = 0000, valore impostato = 10 s
Modalità BCD	Il simbolo # indica il valore dell'istruzione (valore BCD quando si utilizza la modalità BCD).	Da #0000 a #9999 oppure Da #00000000 a #99999999	TIM 0000 #0010
Modalità binaria	Il simbolo & indica un valore decimale.	Da &0 a &65535 oppure Da &0 a &4294967295	TIMX 0000 &10
	Il simbolo # indica il valore dell'istruzione (valore esadecimale quando si utilizza la modalità BCD).	Da #0000 a #FFFF oppure Da #0000 a #FFFFFFFF	TIMX 0000 #A

Nota Quando si utilizza CX-Programmer in modalità BCD o binaria, se il valore numerico viene immesso senza il simbolo di ingresso/visualizzazione # o & indicante la costante, il valore impostato del temporizzatore/contatore verrà interpretato come indirizzo. Se ad esempio si immette TIM 0000 0010, verrà utilizzato come valore impostato il valore nel canale CIO 0010.

6-4-5 Restrizioni

- Non è possibile utilizzare contemporaneamente le modalità BCD e binaria sulla medesima CPU.
- Quando si utilizza la Console di programmazione per creare un nuovo programma utente o per cancellare la memoria, la modalità di aggiornamento dei valori attuali di temporizzatore/contatore viene impostata su BCD.
- Quando si utilizza CX-Programmer versione 3.0 per portare in linea la CPU, viene automaticamente utilizzato il valore impostato memorizzato nella memoria utente del Modulo per la modalità di aggiornamento dei valori attuali di temporizzatore/contatore. Se l'impostazione a livello di CPU differisce da quella configurata per il progetto in CX-Programmer, si verificherà un errore e la connessione in linea non sarà possibile. Viene visualizzato il messaggio riportato di seguito.

Scegliere se modificare l'impostazione della CPU adeguandola a quella del progetto di CX-Programmer o viceversa.

- CX-Programmer versione 2.1 o precedente non consente di leggere i programmi utente della CPU impostati in modalità binaria, ma consente di leggere quelli impostati in modalità BCD.

- Le differenze di funzionamento tra CX-Programmer e la Console di programmazione in caso di immissione di un'istruzione errata per la modalità di aggiornamento dei valori attuali di temporizzatore/contatore sono le seguenti:
  - CX-Programmer:
    se si immette un'istruzione associata a una modalità diversa da quella impostata come modalità di aggiornamento dei valori attuali di temporizzatore/contatore nelle *proprietà del PLC*, si verifica un errore. Ad esempio, quando il PLC è impostato per la modalità binaria nel progetto, si verificherà un errore se si immette TIM come codice mnemonico. Quando è impostata la modalità BCD, si verificherà un errore se si immette TIMX come codice mnemonico.
  - Console di programmazione:
    quando si immette un codice funzione per un'istruzione associata a una modalità diversa da quella impostata come modalità di aggiornamento dei valori attuali di temporizzatore/contatore nella CPU, il codice mnemonico viene automaticamente modificato in base all'impostazione della CPU.

6-4-6 Istruzioni e operandi

Istruzioni

Tipo di istruzione	Nome	Codice mnemonico	
		Modalità BCD	Modalità binaria
Istruzioni di temporizzatore e contatore	TIMER (100 ms)	TIM	TIMX(550)
	HIGH-SPEED TIMER (10 ms)	TIMH(015)	TIMHX(551)
	ONE-MS TIMER (1 ms)	TMHH(540)	TMHHX(552)
	ACCUMULATIVE TIMER (100 ms)	TTIM(087)	TTIMX(555)
	LONG TIMER (100 ms)	TIML(542)	TIMLX(553)
	MULTI-OUTPUT TIMER (100 ms)	MTIM(543)	MTIMX(554)
	COUNTER	CNT	CNTX(546)
	REVERSIBLE COUNTER	CNTR(012)	CNTRX(548)
	RESET TIMER/ COUNTER	CNR(545)	CNRX(547)
Istruzioni di programmazione a blocchi	TIMER WAIT (100 ms)	TIMW(813)	TIMWX(816)
	HIGH-SPEED TIMER WAIT (10 ms)	TMHW(815)	TMHWX(817)
	COUNTER WAIT	CNTW(814)	CNTWX(818)

Istruzioni e operandi

Istruzioni di temporizzatore e contatore

TIMER (100 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TIM	TIM**X**(550)
S (valore impostato del temporizzatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,1 s)	Da 0 a 999,9 s	Da 0 a 6.553,5 s

HIGH-SPEED TIMER (10 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TIMH(015)	TIMH**X**(551)
S (valore impostato del temporizzatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,01 s)	Da 0 a 99,99 s	Da 0 a 655,35 s

ONE-MS TIMER (1 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TMHH(540)	TMHH**X**(552)
S (valore impostato del temporizzatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,001 s)	Da 0 a 9,999 s	Da 0 a 65,535 s

ACCUMULATIVE TIMER (100 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TTIM(087)	TTIM**X**(555)
S (valore impostato del temporizzatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,1 s)	Da 0 a 999,9 s	Da 0 a 6.553,5 s

LONG TIMER (100 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TIML(542)	TIMLX(553)
S, S+1 (valori impostati del temporizzatore)	Da #00000000 a #99999999 (BCD)	Da &0 a &4294967295 (decimale) o da #0000 a #FFFFFFFF (esadecimale)
Tempo di impostazione (unità: 0,1 s)	Da 0 a 999,9 s	Da 0 a 6.553,5 s

MULTI-OUTPUT TIMER (100 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	MTIM(543)	MTIM**X**(554)
Da S a S-7 (valore impostato di ciascun temporizzatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,1 s)	Da 0 a 999,9 s	Da 0 a 6.553,5 s

COUNTER

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	CNT	CNT**X**(546)
S (valore impostato del contatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Impostazione	Da 0 a 9.999 volte	Da 0 a 65.535 volte

REVERSIBLE COUNTER

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	CNTR(012)	CNTR**X**(548)
S (valore impostato del contatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Impostazione	Da 0 a 9.999 volte	Da 0 a 65.535 volte

RESET TIMER/COUNTER

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	CNR(545)	CNR**X**(547)

Istruzioni di programmazione a blocchi

TIMER WAIT (100 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TIMW(813)	TIMW**X**(816)
S (valore impostato del temporizzatore)	#0000 o #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,1 s)	Da 0 a 999,9 s	Da 0 a 6.553,5 s

HIGH-SPEED TIMER WAIT (10 ms)

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	TMHW(815)	TMHW**X**(817)
S (valore impostato del temporizzatore) Unità di misura: 0,01 s	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Tempo di impostazione (unità: 0,01 s)	Da 0 a 999,9 s	Da 0 a 655,35 s

COUNTER WAIT

Nome dell'istruzione	Modalità BCD	Modalità binaria
Codice mnemonico	CNTW(814)	CNTW**X**(818)
S (valore impostato del contatore)	Da #0000 a #9999 (BCD)	Da &0 a &65535 (decimale) o da #0000 a #FFFF (esadecimale)
Impostazione	Da 0 a 9.999 volte	Da 0 a 65.535 volte

6-5 Utilizzo di un interrupt programmato come temporizzatore ad alta precisione (solo CJ1M)

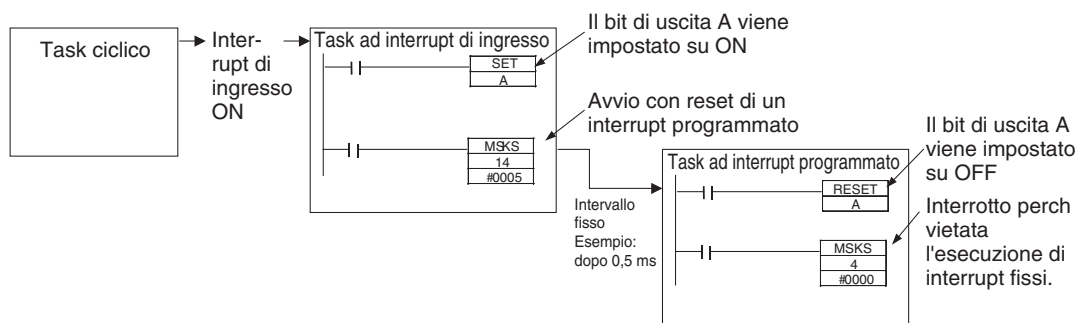
Quando si utilizza una CPU CJ1M, le funzioni riportate di seguito consentono di utilizzare un interrupt programmato come temporizzatore ad alta precisione.

- È possibile specificare unità di 0,1 ms per il temporizzatore fornito dall'interrupt programmato (temporizzatore con intervallo ad alta precisione).
- È possibile eseguire il ripristino, ovvero il riavvio, tramite l'istruzione MSKS(690) (tempo fisso al primo interrupt).
- È possibile leggere i valori attuali del temporizzatore interno tramite l'istruzione MSKR(692) (lettura del valore attuale del temporizzatore di intervallo).

Queste funzioni consentono la creazione di applicazioni, come quella illustrata nell'esempio seguente che fornisce un temporizzatore a singola esecuzione ad alta precisione, in cui l'attivazione del bit di ingresso funge da segnale di avvio, determinando l'attivazione del bit di uscita e la successiva disattivazione dopo un intervallo prefissato.

Esempio

1,2,3...

1. Il task ad interrupt di ingresso viene avviato quando viene attivato il bit di ingresso integrato.
2. Il bit di uscita A viene impostato su ON nel task ad interrupt di ingresso e viene eseguita l'istruzione MSKS(690) per l'avvio con reset di un interrupt programmato.
3. Dopo un intervallo prefissato, il task ad interrupt programmato viene avviato, il bit di uscita A nel task ad interrupt programmato viene impostato su OFF e viene eseguita l'istruzione MSKS(690) per impedire un interrupt programmato.

6-5-1 Impostazione del tempo di interrupt programmato in unità di 0,1 ms

Il tempo di interrupt programmato viene impostato utilizzando l'impostazione per le unità di tempo degli interrupt programmati nella configurazione del PLC e l'istruzione MSKS(690).

Con le CPU CJ1M è possibile impostare il tempo di interrupt programmato in unità di 0,1 ms tra un intervallo minimo di 0,5 ms e un intervallo massimo di 999,9 ms.

Configurazione del PLC

Elemento	Indirizzo del PLC		Valore impostato	Valore predefinito	Frequenza di aggiornamento
	Canale	Bit			
Impostazione dell'unità di tempo degli interrupt programmati	195	Da 00 a 03	0 esadecimale: unità di 10 ms 1 esadecimale: unità di 1 ms 2 esadecimale: unità di 0,1 ms (solo CPU CJ1M)	0 esadecimale	All'avvio

6-5-2 Specifica di un avvio con reset tramite MSKS(690)

Quando si utilizzano le CPU CJ1M e viene eseguita l'istruzione MSKS(690) per avviare l'interrupt programmato, è possibile ripristinare il temporizzatore interno prima di avviare l'interrupt (operazione definita avvio con reset).

Questo metodo viene utilizzato per specificare il tempo al primo interrupt senza dover ricorrere all'istruzione CLI(691).

Gli interrupt programmati vengono avviati utilizzando l'istruzione MSKS(690) per impostare il tempo di interrupt programmato (intervallo tra due interrupt). Dopo l'esecuzione dell'istruzione MSKS(690), tuttavia, il tempo necessario prima dell'avvio del primo task ad interrupt programmato (tempo di avvio del primo interrupt) viene fissato solo se si specifica l'istruzione CLI(691). Pertanto, le CPU CJ1M forniscono una funzione di avvio con reset del temporizzatore interno, che consente di impostare il tempo al primo interrupt senza l'impiego dell'istruzione CLI(691).

Operando dell'istruzione MSKS(690) (solo quando viene specificato un interrupt programmato)

Operando	Valore impostato
N (ID dell'interrupt)	4: interrupt programmato 0, impostazione normale (temporizzatore interno non ripristinato)
	5: interrupt programmato 1, impostazione normale (temporizzatore interno non ripristinato)
	14: interrupt programmato 0, specifica l'avvio con reset (solo CPU CJ1M)
	15: interrupt programmato 1, specifica l'avvio con reset (solo CPU CJ1M)

6-5-3 Lettura del valore attuale del temporizzatore interno con MSKR(692)

Le CPU CJ1M consentono di leggere il valore attuale del temporizzatore interno che misura il tempo di interrupt programmato. Il tempo viene letto dal punto di avvio dell'interrupt programmato o dal punto dell'interrupt programmato precedente. Il valore attuale del temporizzatore interno viene letto eseguendo l'istruzione MSKR(692). L'unità di tempo dipende dall'impostazione dell'unità di tempo degli interrupt programmati definito nella configurazione del PLC, analogamente al tempo di interrupt programmato.

Operandi dell'istruzione MSKR(692) (solo quando viene specificato un interrupt programmato)

Operando	Valore impostato
N (ID dell'interrupt)	4: interrupt programmato 0, legge il tempo di interrupt programmato (valore impostato)
	5: interrupt programmato 1, legge il tempo di interrupt programmato (valore impostato)
	14: interrupt programmato 0, legge il valore attuale del temporizzatore interno (solo CPU CJ1M)
	15: interrupt programmato 1, legge il valore attuale del temporizzatore interno (solo CPU CJ1M)

6-6 Impostazioni di avvio e manutenzione

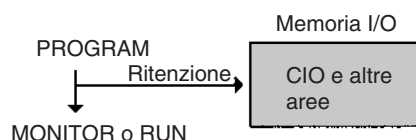
In questa sezione vengono descritte le seguenti funzioni correlate all'avvio e alla manutenzione:

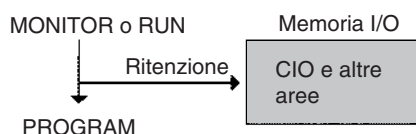
- Funzioni di avvio e arresto a caldo
- Impostazione della modalità di avvio
- Impostazione del ritardo di rilevamento spegnimento
- Disabilitazione degli interrupt di spegnimento
- Uscita RUN
- Funzioni di orologio
- Protezione del programma
- Monitoraggio e programmazione remoti
- Memoria flash
- Impostazione delle condizioni di avvio

6-6-1 Funzioni di avvio e arresto a caldo

Cambiamento della modalità operativa

Avvio a caldo

Impostare su ON il bit di ritenzione IOM (A50012) per conservare tutti i dati* della memoria I/O quando la CPU passa dalla modalità PROGRAM alla modalità RUN/MONITOR per avviare l'esecuzione del programma.

Arresto a caldo

Quando il bit di ritenzione IOM (A50012) è impostato su ON, tutti i dati* della memoria I/O vengono conservati anche quando la CPU passa dalla modalità RUN/MONITOR alla modalità PROGRAM per interrompere l'esecuzione del programma.

Nota *Durante il cambio di modalità operativa (PROGRAM ↔ RUN/MONITOR), le seguenti aree della memoria I/O vengono cancellate, a meno che il bit di ritenzione IOM non sia impostato su ON: area CIO (area degli I/O, area dei data link, area del Modulo CPU bus, area del Modulo di I/O speciale, area della scheda interna, area SYSMAC BUS, area dei terminali di I/O, area DeviceNet (CompoBus/D) e aree I/O interne), area di lavoro, flag di completamento dei temporizzatori e valori attuali dei temporizzatori. Le aree relative alla scheda interna, a SYSMAC BUS e ai terminali di I/O sono supportati solo dalle CPU della serie CS.

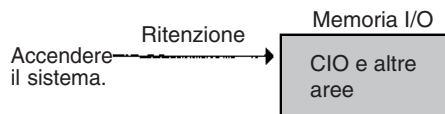
Flag e canali dell'area ausiliaria

Nome	Indirizzo	Descrizione
bit di ritenzione IOM	A50012	Se questo bit è impostato su ON, tutti i dati della memoria I/O vengono conservati quando la modalità operativa cambia (PROGRAM ↔ RUN/MONITOR).

Se il bit di ritenzione IOM è impostato su ON, tutte le uscite dei Moduli di uscita vengono conservate quando l'esecuzione del programma viene interrotta. Quando il programma viene nuovamente avviato, le uscite si troveranno nel medesimo stato attivo al momento dell'interruzione del programma.

Se il bit di ritenzione IOM è impostato su OFF, le istruzioni verranno eseguite dopo la cancellazione delle uscite.

Accensione del PLC

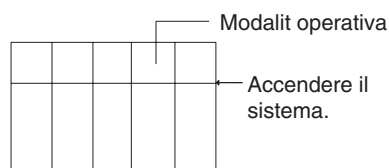
Affinché tutti i dati* della memoria I/O vengano conservati all'accensione del PLC (OFF → ON), è necessario impostare su ON e proteggere il bit di ritenzione IOM nella configurazione del PLC (indirizzo 80, stato del bit di ritenzione IOM all'accensione).

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Descrizione
bit di ritenzione IOM	A50012	Se questo bit è impostato su ON, tutti i dati della memoria I/O vengono conservati quando la modalità operativa cambia (PROGRAM ↔ RUN/MONITOR).

Configurazione del PLC

Indirizzo della Console di programmazione	Nome	Impostazione	Valore predefinito
80 bit 15	Stato del bit di ritenzione IOM all'accensione	0: il bit di ritenzione IOM viene azzerato all'accensione. 1: il bit di ritenzione IOM viene mantenuto all'accensione.	0 (azzerato)

6-6-2 Impostazione della modalità di avvio

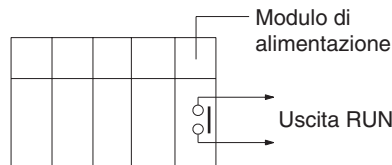
È possibile impostare la modalità operativa iniziale della CPU, ovvero all'accensione della CPU, nella configurazione del PLC.

Configurazione del PLC

Indirizzo della Console di programmazione	Nome	Descrizione	Impostazione	Valore predefinito
81	Modalità di avvio	Specifica la modalità operativa da utilizzare all'avvio.	PRCN: modalità operativa impostata sul selettore di modalità della Console di programmazione PRG: Modalità PROGRAM MON: Modalità MONITOR RUN: modalità RUN	PRCN: modalità operativa imostata sul selettore di modalità della Console di programmazione

Nota Se si imposta la modalità di avvio su PRCN (modalità operativa impostata sul selettore di modalità della Console di programmazione) ma non è collegata alcuna Console di programmazione, la CPU verrà avviata in modalità RUN. Si consiglia di modificare il valore predefinito della configurazione del PLC per l'avvio in modalità MONITOR o PROGRAM all'accensione. Si tenga presente

che, nelle medesime condizioni, le CPU CS1 della serie CS verranno avviate in modalità PROGRAM.

6-6-3 Uscita RUN

Alcuni Moduli di alimentazione (C200HW-PA204R, C200HW-PA209R, CJ1W-PA205R e CS1D-PA207R) dispongono di un'uscita RUN. Questo punto di uscita è attivato (chiuso) quando la CPU funziona in modalità RUN o MONITOR e disattivato (aperto) quando la CPU è in modalità PROGRAM.

È possibile utilizzare l'uscita RUN per creare circuiti di sicurezza esterni, ad esempio un circuito di arresto di emergenza che impedisce l'alimentazione esterna del Modulo di uscita quando il PLC non è acceso.

Nota Se si utilizza un Modulo di alimentazione privo di uscita RUN, è possibile creare un'uscita equivalente programmando il flag di attivazione permanente (A1) come condizione di esecuzione per un punto di uscita da un Modulo di uscita.

⚠ **Attenzione** Se l'alimentazione esterna del Modulo di uscita entra in funzione prima dell'alimentazione del PLC, è possibile che si riscontri un temporaneo funzionamento incorretto del Modulo di uscita all'accensione del PLC. Per evitare ciò, aggiungere un circuito esterno che impedisca alla fonte di alimentazione esterna di erogare corrente al Modulo di uscita prima che venga alimentato il PLC. Creare un circuito di sicurezza come quello sopra descritto per garantire che la fonte di alimentazione esterna eroghi corrente solo quando il PLC è in funzione in modalità RUN o MONITOR.

6-6-4 Impostazione del ritardo di rilevamento spegnimento

Di norma una caduta di tensione viene rilevata entro un periodo compreso tra 10 e 25 ms (2 e 5 ms per gli alimentatori a corrente continua) dopo il calo della tensione al di sotto dell'85% del valore nominale minimo (80% per gli alimentatori a corrente continua). Nella configurazione del PLC è disponibile un'impostazione (indirizzo 225, bit da 0 a 7, ritardo di rilevamento spegnimento) che consente di estendere questo periodo di tempo di ulteriori 10 ms (2 ms per gli alimentatori a corrente continua).
Quando è abilitato, il task ad interrupt di spegnimento viene eseguito alla conferma della caduta di tensione. In caso contrario, la CPU viene reimpostata e l'operazione interrotta.

Impostazioni correlate

Indirizzo	Nome	Descrizione	Impostazione	Valore predefinito
CIO 256, bit da 00 a 07	Ritardo di rilevamento spegnimento	Imposta il ritardo relativo al rilevamento di una caduta di tensione.	Da 00 a 0A (esadecimale): da 0 a 10 ms	00 (esadecimale): 0 ms

6-6-5 Disabilitazione degli interrupt di spegnimento

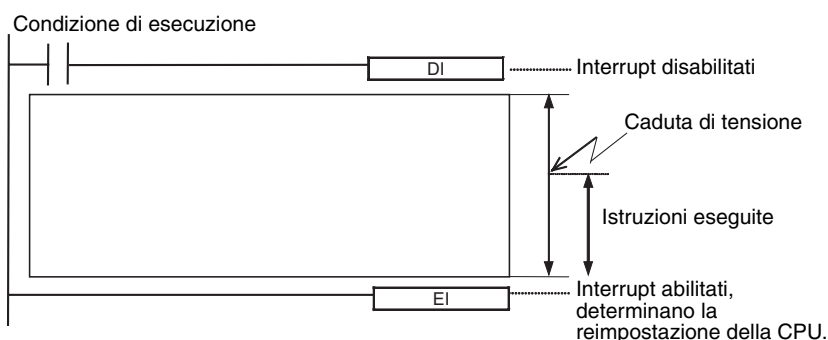
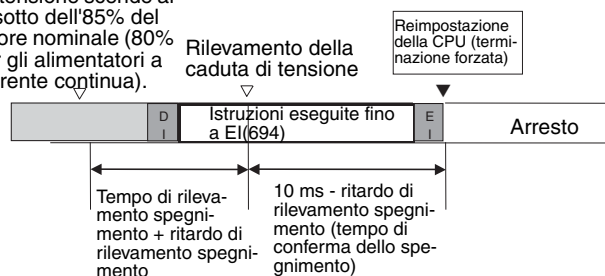
Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Nelle CPU CS1-H, CJ1-H, CJ1M o CS1D è possibile proteggere aree del programma dagli interrupt di spegnimento in modo che vengano comunque

eseguite anche quando l'alimentazione viene interrotta. A tale scopo, è possibile utilizzando le istruzioni DISABLE
INTERRUPTS (DI(693)) e ENABLE INTERRUPTS (EI(694)).

Questa funzione può essere utilizzata con set di istruzioni che devono essere eseguite in gruppo, ad esempio per garantire che all'accensione successiva l'esecuzione non venga avviata utilizzando dati intermedi.

Procedura

1,2,3...
1. Impostare la disabilitazione dell'impostazione per gli interrupt di spegnimento in A530 su A5A5 esadecimale per disabilitare gli interrupt.
2. Attivare la disabilitazione degli interrupt di spegnimento nella configurazione del PLC (impostazione predefinita).
3. Utilizzare l'istruzione DI(693) per disabilitare gli interrupt posizionati prima della sezione del programma che si desidera proteggere, quindi utilizzare l'istruzione EI(694) per abilitare gli interrupt posizionati dopo tale sezione. Tutte le istruzioni tra DI(693) e EI(694) verranno completate prima dell'esecuzione dell'interrupt di spegnimento, anche se durante l'esecuzione di tali istruzioni si verifica una caduta di tensione.

Impostazioni correlate

Nome	Indirizzo	Descrizione
Disabilitazione dell'impostazione per gli interrupt di spegnimento	A530	Abilita l'utilizzo dell'istruzione DI(693) per disabilitare l'elaborazione dell'interrupt di spegnimento (fatta eccezione per l'esecuzione del task ad interrupt di spegnimento) fino all'esecuzione di EI(694). A5A5 esadecimale: abilita l'utilizzo dell'istruzione DI(693) per la disabilitazione dell'elaborazione dell'interrupt di spegnimento Un altro valore: disabilita l'utilizzo dell'istruzione DI(693) per la disabilitazione dell'elaborazione dell'interrupt di spegnimento

6-6-6 Funzioni di orologio

I PLC della serie CS/CJ sono dotati delle seguenti funzioni di orologio:

- Monitoraggio dell'ora in cui si sono verificate cadute di tensione

- Monitoraggio dell'ora dall'accensione del PLC
- Monitoraggio del tempo totale di esercizio del PLC

Nota Poiché le CPU CS1 della serie CS vengono fornite senza batteria di backup, quando si installa la batteria, la data e l'ora dell'orologio interno della CPU vengono riportate a 00/01/01 00:00:00 o un altro valore. Per utilizzare le funzioni dell'orologio, installare la batteria, accendere il sistema e impostare la data e l'ora mediante un dispositivo di programmazione (Console di programmazione o CX-Programmer) oppure tramite l'apposito comando FINS (07 02, CLOCK WRITE). L'orologio interno della CPU verrà avviato subito dopo l'impostazione.

Flag e canali dell'area ausiliaria

Nome	Indirizzi	Funzione
Dati dell'orologio	Da A35100 ad A35107	Secondi: da 00 a 59 (BCD)
	Da A35108 ad A35115	Minuti: da 00 a 59 (BCD)
	Da A35200 ad A35207	Ora: da 00 a 23 (BCD)
	Da A35208 ad A35215	Giorno del mese: da 00 a 31 (BCD)
	Da A35300 ad A35307	Mese: da 00 a 12 (BCD)
	Da A35308 ad A35315	Anno: da 00 a 99 (BCD)
	Da A35400 ad A35407	Giorno della settimana: 00: domenica, 01: lunedì, 02: martedì, 03: mercoledì, 04: giovedì, 05: venerdì, 06: sabato
Ora di avvio	A510 e A511	Indica l'ora di accensione del sistema.
Ora caduta di tensione	A512 e A513	Indica l'ora in cui si è verificata l'ultima caduta di tensione.
Tempo di esercizio totale	A523	Indica per quanto tempo (valore binario) il PLC è rimasto accesso, espresso in unità di 10 ore.

Istruzioni correlate

Istruzione	Nome	Funzione
SEC(065)	HOURS TO SECONDS	Converte i dati temporali in formato ore/minuti/secondi in un tempo equivalente espresso solo in secondi.
HMS(066)	SECONDS TO HOURS	Converte i dati temporali espressi in secondi in un tempo equivalente in formato ore/minuti/secondi.
CADD(730)	CALENDAR ADD	Aggiunge tempo ai dati di calendario nei canali specificati.
CSUB(731)	CALENDAR SUBTRACT	Sottrae tempo dai dati di calendario nei canali specificati.
DATE(735)	CLOCK ADJUSTMENT	Modifica l'impostazione dell'orologio interno in base all'impostazione dei canali sorgente specificati.

6-6-7 Protezione del programma

È possibile proteggere da scrittura e in modo completo (protezione da lettura e scrittura) un programma utente creato per i Moduli della serie CS/CJ.

Protezione da scrittura mediante DIP switch

È possibile proteggere da scrittura il programma utente posizionando su ON il pin 1 del DIP switch della CPU. Quando il pin è posizionato su ON, non è

possibile modificare il programma utente da un dispositivo di programmazione, incluse le Console di programmazione. Questa funzione consente di impedire la sovrascrittura involontaria del programma.

È comunque possibile leggere e visualizzare un programma protetto da scrittura.

Nota Per le CPU della serie CS/CJ versione 2.0 o successiva è possibile utilizzare l'impostazione del DIP switch sopra riportata o, nel caso in cui venga utilizzato CX-Programmer versione 4.0 o successiva, è possibile impostare la protezione opzionale da lettura/scrittura del programma quando si imposta una password per l'intero programma o per uno o più task. Per ulteriori informazioni, fare riferimento ad *Abilitazione/Disabilitazione della creazione di file di programma nella memoria file* nella sezione *1-4-2 Migliore protezione da lettura mediante password* del *Manuale dell'operatore della serie CJ*.

Protezione da lettura/scrittura mediante password

È possibile bloccare l'accesso sia in lettura che in scrittura all'area del programma utente da CX-Programmer. La protezione del programma impedisce la copia non autorizzata del programma e la conseguente perdita dei diritti d'autore. L'impostazione di una password per la protezione del programma tramite un dispositivo di programmazione impedisce l'accesso all'intero programma.

Nota

1. Se si dimentica la password, non sarà possibile trasferire il programma dal PLC al computer. Si consiglia pertanto di annotare la password e conservarla in un luogo sicuro.
2. Se si dimentica la password, non sarà possibile trasferire il programma dal computer al PLC. È possibile trasferire i programmi dal computer al PLC anche se la password di protezione non è stata rilasciata.

Protezione tramite password

1,2,3...

1. Registrare una password in linea o non in linea nel seguente modo:
   a) Selezionare il PLC e scegliere ***Properties*** (Proprietà) dal menu View (Visualizza).
   b) Selezionare ***Protection*** (Protezione) nella finestra di dialogo PLC Properties (Proprietà del PLC) e immettere la password.
2. Impostare la protezione tramite password in linea nel seguente modo:
   a) Selezionare ***PLC, Protection*** (PLC, Protezione), quindi fare clic su ***Set*** (Imposta). Viene visualizzata la finestra di dialogo Protection Setting (Impostazione protezione).
   b) Fare clic su **OK**.

Nota Per le CPU della serie CS/CJ versione 2.0 o successiva è possibile impostare la protezione da lettura non solo per l'intero programma, ma anche per task specifici. Per ulteriori informazioni, fare riferimento a *Protezione da lettura per singoli task mediante password* nella sezione *1-4-2 Migliore protezione da lettura mediante password* del *Manuale dell'operatore della serie CJ*.

Conferma della data di creazione del programma utente

Nelle CPU CS1-H, CJ1-H, CJ1M o CS1D, è possibile verificare le date di creazione del programma e dei parametri controllando il contenuto degli indirizzi da A090 ad A097.

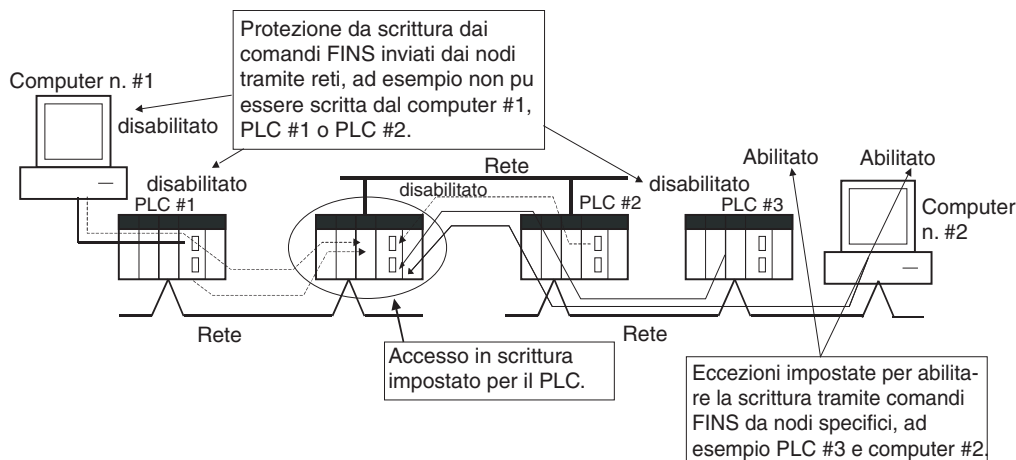
Canali dell'area ausiliaria

Nome	Indirizzo	Descrizione	
data del programma utente	Da A090 ad A093	La data e l'ora dell'ultima sovrascrittura del programma utente sono espresse in formato BCD.	
		Da A09000 ad A09007	Secondi (da 00 a 59 in formato BCD)
		Da A09008 ad A09015	Minuti (da 00 a 59 in formato BCD)
		Da A09100 ad A09107	Ora (da 00 a 23 in formato BCD)
		Da A09108 ad A09115	Giorno del mese (da 01 a 31 in formato BCD)
		Da A09200 ad A09207	Mese (da 01 a 12 in formato BCD)
		Da A09208 ad A09215	Anno (da 00 a 99 in formato BCD)
		Da A09300 ad A09307	Giorno (da 00 a 06 in formato BCD) Giorno della settimana: 00: domenica, 01: lunedì, 02: martedì, 03: mercoledì, 04: giovedì, 05: venerdì, 06: sabato
data dei parametri	Da A094 ad A097	La data e l'ora dell'ultima sovrascrittura dei parametri sono espresse in formato BCD. Il formato è identico a quello della data del programma utente descritto in precedenza.	

6-6-8 Protezione da scrittura dai comandi FINS inviati alle CPU tramite reti

Per le CPU della serie CS/CJ versione 2.0 o successiva è possibile impostare la protezione per impedire l'esecuzione di operazioni di scrittura e controllo delle CPU tramite i comandi FINS attraverso la rete, ovvero connessioni diverse dalla connessioni seriali dirette. Ciò include la scrittura da applicazioni che utilizzano FinsGateway, nonché da CX-Programmer, CX-Protocol e CX-Process. In questo caso, la lettura è ancora consentita.

Se si imposta la protezione da scrittura, verrà impedita l'esecuzione delle seguenti operazioni: scaricamento del programma utente, della configurazione del PLC, della memoria I/O o di altri dati, modifica della modalità operativa, modifica in linea e qualsiasi altra operazione di scrittura o controllo.

Anche se si imposta la protezione da scrittura, è possibile specificare determinati nodi come eccezioni e consentire l'esecuzione delle operazioni di scrittura e di controllo da essi.

Nota questa funzione consente di impedire le operazioni di scrittura e controllo solo per i comandi FINS, ma non ha effetto sulle operazioni di scrittura e controllo eseguite con altri metodi, ad esempio tramite data link.

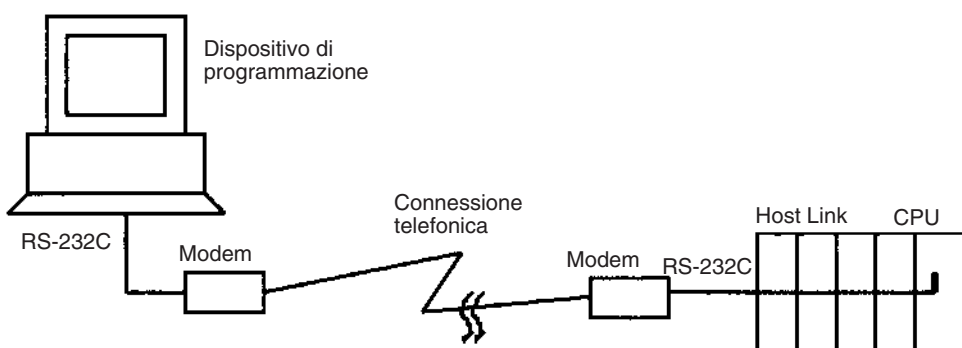
Per ulteriori informazioni, fare riferimento alla sezione *1-4-3 Protezione da scrittura dai comandi FINS inviati alle CPU tramite reti* nel *Manuale dell'operatore della serie CJ*.

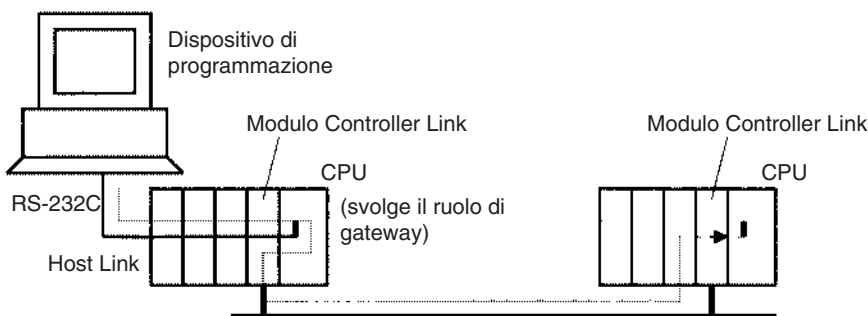
6-6-9 Monitoraggio e programmazione remoti

È possibile programmare e monitorare i PLC della serie CS/CJ in modalità remota tramite un modem o una rete Controller Link.

1,2,3...

1. Connessioni modem

   L'uso combinato della funzione Host Link e di un modem consente di monitorare il funzionamento di un PLC distante e di modificarne il programma via telefono, nonché di trasferire dati. Questo tipo di connessioni supporta tutte le operazioni in linea del dispositivo di programmazione.

2. Connessioni di rete Controller Link

   I PLC collegati a una rete Controller Link o Ethernet possono essere programmati e monitorati tramite Host Link. Questo tipo di connessioni supporta tutte le operazioni in linea del dispositivo di programmazione.

6-6-10 Profili dei Moduli

Con CX-Programmer è possibile reperire le informazioni relative ai Moduli della serie CS/CJ elencate di seguito.

- Informazioni di produzione (numero di lotto, numero di serie e così via): agevolano la ricerca delle informazioni da fornire a OMRON in caso di problemi relativi al Modulo.
- Informazioni sul modulo (tipo, codice del modello, posizione nel sistema o nello slot corretta): consentono di ottenere facilmente informazioni sul montaggio.
- Testo definito dall'utente (massimo 256 caratteri): consente di registrare le informazioni necessarie per la manutenzione (cronologia dell'ispezione

del Modulo, numero della linea di produzione e altre informazioni sull'applicazione) nelle schedine di memoria.

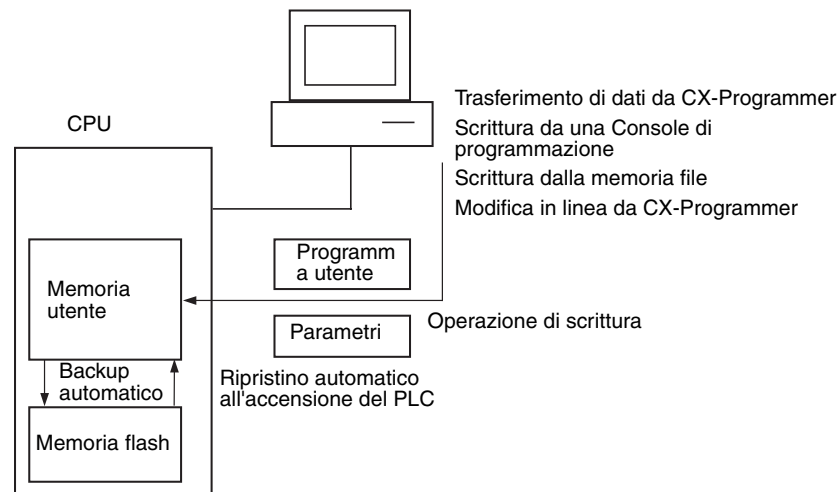
6-6-11 Memoria flash

Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Le CPU CS1-H, CJ1-H, CJ1M e CS1D eseguono automaticamente il backup del programma utente e dei parametri nella memoria flash quando vengono scritti o modificati nella CPU.

Viene eseguito il backup automatico dei seguenti dati: programma utente e parametri, inclusi la configurazione del PLC, tabelle I/O registrati, tabelle di routing e dati del Modulo CPU bus, quali le tabelle di data link.

Il backup automatico dei dati viene eseguito ogni volta che il programma utente o i parametri vengono scritti nella CPU. Ciò comprende le operazioni di trasferimento dei dati da CX-Programmer, la scrittura dei dati da una Console di programmazione, la modifica in linea, il trasferimento dei dati da una schedina di memoria o dalla memoria file nell'area EM e così via.

Il programma utente e i dati dei parametri scritti nella memoria flash vengono automaticamente trasferiti nella memoria utente della CPU all'avvio.

Nota

1. Durante la scrittura di dati nella memoria flash, l'indicatore BKUP posto sulla parte anteriore della CPU si illumina. Non spegnere la CPU fino al termine dell'operazione di backup, ovvero finché l'indicatore BKUP non si sarà spento, dopo il trasferimento di dati da un dispositivo di programmazione o dalla memoria file o dopo l'esecuzione delle modifiche in linea.
2. Solo nel caso della modifica in linea e solo se nella CPU è installata una batteria, la CPU verrà riavviata ripristinando la condizione precedente (ad esempio, con l'indicatore BKUP acceso) anche se l'alimentazione viene interrotta prima del completamento dell'operazione di backup. In questo caso, l'avvio della CPU può richiedere fino a 1 minuto. Se la CPU deve restare spenta per un lungo periodo di tempo, prima di spegnerla sarà comunque necessario accertarsi che l'operazione di backup sia terminata, anche se la CPU è dotata di batteria.

La quantità di tempo necessaria per il backup dei dati (il tempo in cui l'indicatore BKUP rimane acceso) dipende dalle dimensioni del programma utente, come indicato nella tabella che segue.

Dimensioni del programma utente	Tempo di elaborazione del backup		
	Modalità MONITOR		Modalità PROGRAM
	Tempo di ciclo di 0,4 ms (esempio)	Tempo di ciclo di 10,0 ms (esempio)	
10.000 step	2 s	8 s	1 s
60.000 step	11 s	42 s	6 s
250.000 step	42 s	170 s	22 s

Nota
1. L'indicatore BKUP si illumina quando la CPU è alimentata.
2. A seconda del tipo di modifica in linea eseguita, il backup dei dati può richiedere fino a 1 minuto.

⚠ **Attenzione** Le CPU CS1-H, CJ1-H, CJ1M e CS1D eseguono automaticamente il backup del programma utente e dei dati di parametro nella memoria flash quando tali dati vengono scritti nella CPU. I dati della memoria I/O (comprese le aree DM, EM e HR) non vengono tuttavia scritti nella memoria flash. Un'apposita batteria consente di conservare i dati delle aree DM, EM e HR nel caso in cui si verifichino cadute di tensione. Se si verificano errori relativi alla batteria, i dati contenuti in queste aree potrebbero risultare non corretti dopo la caduta di tensione. Se i dati delle aree DM, EM e HR vengono utilizzati per il controllo di uscite esterne, impedire l'invio di dati non corretti quando il flag di errore della batteria (A40204) è attivato.

Nota Quando si esegue il backup di dati da CX-Programmer per operazioni di trasferimento diverse dai consueti trasferimenti di dati, eseguiti di norma facendo clic su ***PLC, Transfer*** (PLC, Trasferisci), è possibile visualizzare lo stato del backup nella finestra Memory Backup Status (Stato di backup della memoria). Per visualizzare questa finestra, selezionare la relativa opzione nelle proprietà del PLC, quindi scegliere la finestra dal menu View (Visualizza). Per le normali operazioni di trasferimento, lo stato del backup viene visualizzato nella finestra di trasferimento dopo lo stato del trasferimento del programma e altri dati.

Flag dell'area ausiliaria

Nome	Indirizzo	Descrizione
Flag di errore della memoria flash	A40310	Viene attivato quando si verifica un errore nella memoria flash.

6-6-12 Impostazione delle condizioni di avvio

Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

L'avvio di alcuni Moduli e schede interne richiede un tempo considerevolmente lungo dopo l'accensione e ciò incide sui tempi di avvio della CPU. È possibile impostare la configurazione del PLC in modo che la CPU venga avviata senza attendere l'inizializzazione di tali Moduli.

Questa impostazione è applicabile ai Moduli di interfaccia bus CS1 a rete aperta ITNC-EIS01-CST e ITNC-EIX01-CST. Attualmente non esistono schede interne a cui sia possibile applicare questa impostazione (dati aggiornati a ottobre 2001).

È possibile controllare questa funzione configurando le condizioni di avvio e l'impostazione della scheda interna, come descritto nella tabella che segue.

Condizioni di avvio	Configurazione del PLC	
	Condizione di avvio (indirizzo Console di programmazione 83, bit 15)	**Impostazione della scheda interna (indirizzo Console di programmazione 84, bit 15)**
Per eseguire l'avvio senza attendere l'inizializzazione di tutti i Moduli e le schede	1: abilita l'avvio senza attesa	1: non attende l'inizializzazione di schede interne specifiche
Per eseguire l'avvio senza attendere l'inizializzazione di tutti i Moduli ma attendendo l'inizializzazione delle schede	1: abilita l'avvio senza attesa	0: attende l'inizializzazione di tutte le schede interne prima dell'avvio
Per attendere l'inizializzazione di tutti i Moduli e le schede prima di eseguire l'avvio	0: attende l'inizializzazione di tutti i Moduli e delle schede	Qualsiasi

Nota Le CPU CS1 non vengono avviate finché il processo di avvio di tutti i Moduli e le schede non è completato.

Configurazione del PLC

Indirizzo della Console di programmazione		Nome	Impostazione	Valore predefinito	Frequenza di aggiornamento della CPU
Canale	**Bit**				
83	15	Condizione di avvio	0: attesa dell'inizializzazione di Moduli e schede 1: nessuna attesa	0: attesa	Accendere il sistema.
84	15	Impostazione della scheda interna	0: attesa dell'inizializzazione di tutte le schede interne 1: nessuna attesa per schede specifiche	0: attesa	Accendere il sistema.

Condizione di avvio

0: se uno o più Moduli o schede specifici non ha completato il processo di avvio, la CPU verrà posta in attesa in modalità MONITOR o PROGRAM e attenderà l'avvio di tutti i Moduli e le schede.

1: anche se uno o più Moduli o schede specifici non ha completato il processo di avvio, la CPU proseguirà l'avvio in modalità MONITOR o PROGRAM. Il funzionamento delle schede interne dipende anche dall'impostazione descritta di seguito.

Impostazione della scheda interna

Questa impostazione viene utilizzata solo se la condizione di avvio è impostata su 1 per abilitare l'avvio senza attesa di Moduli e schede specifici. L'impostazione viene ignorata se la condizione di avvio è impostata su 0.

0: se per una o più schede specifiche non ha completato il processo di avvio, la CPU verrà posta in attesa in modalità MONITOR o PROGRAM e attenderà l'avvio di tutte le schede.

1: anche se una o più schede specifiche non ha completato il processo di avvio, la CPU proseguirà l'avvio in modalità MONITOR o PROGRAM.

6-7 Funzioni di diagnostica

Questa sezione fornisce una panoramica delle seguenti funzioni di diagnostica e debug:

- Log degli errori
- Funzione di disattivazione delle uscite
- Funzioni di allarme per guasto (FAL(006) e FALS(007))
- Funzione di individuazione del punto di funzionamento incorretto (FPD(269))

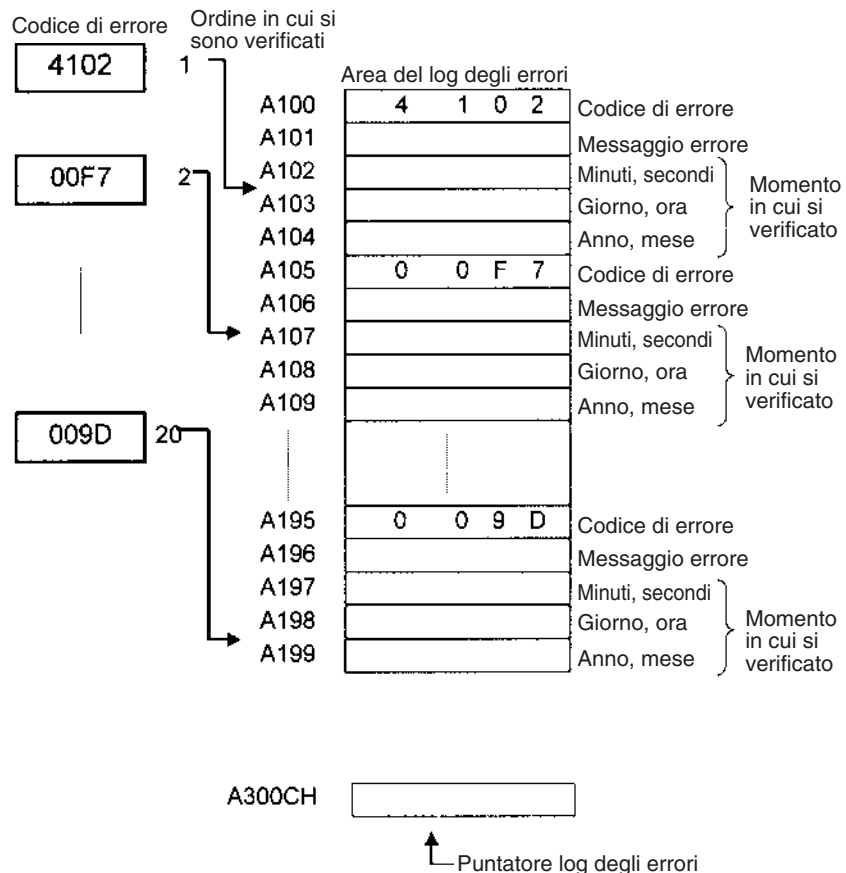
6-7-1 Log degli errori

Ogni volta che si verifica un errore in un PLC della serie CS/CJ, le informazioni relative all'errore vengono memorizzate nella CPU all'interno dell'area del log degli errori. Tali informazioni comprendono il codice di errore (memorizzato in A400), il contenuto e l'ora in cui si è verificato l'errore. È possibile registrare fino a 20 errori nel log.

Oltre agli errori generati dal sistema, il PLC registra gli errori FAL(006) e FALS(007) definiti dall'utente, agevolando in tal modo il recupero di informazioni sullo stato operativo del sistema.

Per ulteriori informazioni, consultare il capitolo sulla risoluzione dei problemi del manuale *CS/CJ Series Operation Manual*.

Nota Quando nel programma viene eseguita l'istruzione FAL(006) o FALS(007), viene generato un errore definito dall'utente. Le condizioni di esecuzione di tali istruzioni costituiscono le condizioni di errore definite dall'utente. FAL(006) genera un errore non fatale, mentre FALS(007) genera un errore fatale che interrompe l'esecuzione del programma.

Quando si verificano più di 20 errori, i dati relativi all'errore meno recente (negli indirizzi da A100 ad A104) vengono eliminati, i 19 record rimanenti vengono spostati verso il basso di una posizione e il record più recente viene memorizzato nell'intervallo di indirizzi da A195 ad A199.

Il numero di record viene memorizzato in formato binario nel puntatore del log degli errori (A300). Il puntatore non viene incrementato quando si sono verificati più di 20 errori.

6-7-2 Funzione di disattivazione delle uscite

Come misura di emergenza in caso di errore, è possibile disattivare tutte le uscite dei Moduli di uscita impostando su ON il bit di disattivazione uscite (A50015). La modalità operativa RUN o MONITOR rimarrà invariata ma verranno disattivate tutte le uscite.

Nota Di norma, ovvero quando il bit di ritenzione IOM è impostato su OFF, tutte le uscite dei Moduli vengono disattivate quando si passa dalla modalità operativa RUN o MONITOR alla modalità PROGRAM. È possibile utilizzare il bit di disattivazione uscite per disattivare tutte le uscite senza passare alla modalità PROGRAM e interrompere l'esecuzione del programma.

Precauzioni di applicazione per DeviceNet

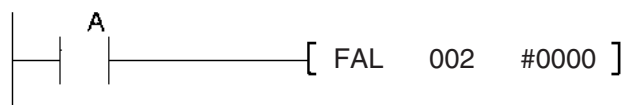
Quando si utilizza la funzione master con i Moduli CS1W-DRM21 o CJ1W-DRM21, vengono disattivate tutte le uscite slave. Quando si utilizza la funzione slave, vengono disattivati tutti gli ingressi al Modulo master. Sul Modulo C200HW-DRM21-V1, tuttavia, le uscite slave non vengono disattivate.

6-7-3 Funzioni di allarme per guasto

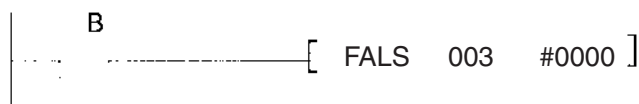
Le istruzioni FAL(006) e FALS(007) generano errori definiti dall'utente. FAL(006) genera un errore non fatale, mentre FALS(007) genera un errore fatale che interrompe l'esecuzione del programma.

Quando si verificano le condizioni di un errore definito dall'utente, ossia le condizioni di esecuzione per FAL(006) o FAL(007), vengono eseguite l'istruzione di allarme per guasto e le operazioni indicate di seguito.

1,2,3...
1. Viene attivato il flag di errore FAL (A40215) o FALS (A40106).
2. Il codice di errore corrispondente viene scritto in A400.
3. Il codice di errore e l'ora in cui l'errore si è verificato vengono memorizzati nel log degli errori.
4. L'indicatore di errore sul pannello frontale della CPU lampeggia o si illumina.
5. Se viene eseguita l'istruzione FAL(006), la CPU continuerà a funzionare. Se viene eseguita l'istruzione FAL(007), la CPU verrà arrestata, ovvero l'esecuzione del programma viene interrotta.

Funzionamento di FAL(006)

Quando si verifica la condizione di esecuzione A, viene generato un errore FAL con numero 2 e vengono attivati il flag di errore FAL (A40215) e il flag di FAL numero 2 (A36002). L'esecuzione del programma continua.

È possibile cancellare gli errori generati da FAL(006) eseguendo FAL(006) con il numero FAL 00 o procedere all'operazione di lettura/cancellazione dell'errore da un dispositivo di programmazione (compresa una Console di programmazione).

Funzionamento di FALS(007)

Quando si verifica la condizione di esecuzione B, viene generato un errore FALS con numero 3 e viene attivato il flag di errore FALS (A40106). L'esecuzione del programma viene interrotta.

È possibile cancellare gli errori generati da FALS(007) eliminando la causa dell'errore ed eseguendo l'operazione di lettura/cancellazione dell'errore da un dispositivo di programmazione (compresa una Console di programmazione).

6-7-4 Individuazione del punto di funzionamento incorretto

FPD(269) esegue il monitoraggio del tempo e un'analisi diagnostica della logica. Se l'uscita diagnostica non viene attivata entro il tempo di monitoraggio specificato, la funzione di monitoraggio del tempo genererà un errore non fatale. La funzioni di analisi diagnostica della logica indica quale ingresso impedisce l'attivazione dell'uscita di diagnostica.

Funzione di monitoraggio del tempo

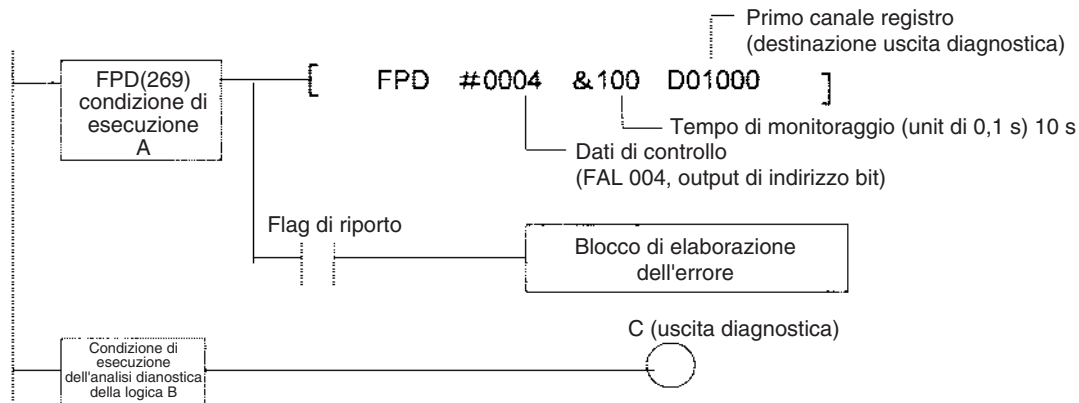
L'istruzione FPD(269) inizia a monitorare il tempo al momento dell'esecuzione e attiva il flag di riporto quando l'uscita di diagnostica non viene attivata entro il tempo di monitoraggio specificato. È possibile programmare il flag di riporto come condizione di esecuzione di un blocco di elaborazione dell'errore. Inoltre, è possibile programmare FPD(269) in modo che generi un errore FAL non fatale con il numero di FAL desiderato.

Quando viene generato un errore FAL, verrà registrato un messaggio preimpostato che potrà essere visualizzato su un dispositivo di programmazione. È possibile impostare l'istruzione FPD(269) in modo da ottenere il risultato dell'analisi diagnostica della logica, ovvero l'indirizzo del bit che impedisce l'attivazione dell'uscita di diagnostica, prima del messaggio.

È possibile utilizzare la funzione di autoimpostazione per determinare in modo automatico il tempo effettivo richiesto per l'attivazione dell'uscita diagnostica e impostare il tempo di monitoraggio.

Funzione di analisi diagnostica della logica

L'istruzione FPD(269) consente di determinare quale bit di ingresso impedisce l'attivazione dell'uscita di diagnostica e registrare l'indirizzo di tale bit. È possibile impostare l'invio del risultato come output di indirizzo bit (indirizzo di memoria del PLC) o come output di messaggio (ASCII).

- Se si seleziona l'output di indirizzo bit, sarà possibile trasferire l'indirizzo di memoria del PLC relativo al bit in un registro indice e fare indirettamente riferimento al registro per un'elaborazione successiva.
- Se si seleziona l'output di messaggio, l'indirizzo del bit verrà registrato in un messaggio ASCII che potrà essere visualizzato su un dispositivo di programmazione.

Monitoraggio del tempo:

Consente di verificare se l'uscita C viene attivata entro 10 secondi dal segnale di ingresso A. Se l'uscita C non viene attivata entro 10 secondi, verrà rilevato un funzionamento incorretto e verrà attivato il flag di riporto. Il flag di riporto esegue il blocco di elaborazione dell'errore. Viene inoltre generato un errore FAL (non fatale) con numero 004.

Analisi diagnostica della logica:

L'istruzione FPD(269) determina quale bit di ingresso del blocco B impedisce l'attivazione dell'uscita C. L'indirizzo di tale bit viene memorizzato in D01000 e D01001.

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Funzionamento
Codice errore	A400	Quando si verifica un errore, il relativo codice viene memorizzato in A400.
Flag di errore FAL	A40215	Viene attivato quando viene eseguita l'istruzione FAL(006).
Flag di errore FALS	A40106	Viene attivato quando viene eseguita l'istruzione FALS(007).
Flag numeri di FAL eseguiti	Da A360 ad A391	Il flag corrispondente viene attivato quando si verifica un errore FAL(006) o FALS(007).
Area del log degli errori	Da A100 ad A199	L'area del log degli errori contiene informazioni sui 20 errori più recenti.
Puntatore log degli errori	A300	Quando si verifica un errore, il contenuto del puntatore viene incrementato di 1 per indicare la posizione in cui verrà registrato il record di errore successivo, specificando il tal modo l'offset dall'inizio dell'area del log degli errori (A100).
Bit di ripristino del puntatore log degli errori	A50014	Attivare questo bit per reimpostare il puntatore del log degli errori (A300) su 00.
Bit di autoimpostazione FPD	A59800	Attivare questo bit per impostare automaticamente il tempo di monitoraggio quando viene eseguita l'istruzione FPD(269).

6-7-5 Simulazione degli errori di sistema

Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

È possibile utilizzare le istruzioni FAL(006) e FALS(007) per generare intenzionalmente errori di sistema fatali e non fatali. Ciò consente di verificare se i messaggi vengono visualizzati correttamente sui terminali programmabili (PT) o altre interfacce durante il debug del sistema.

A tale scopo, attenersi alla procedura descritta di seguito.

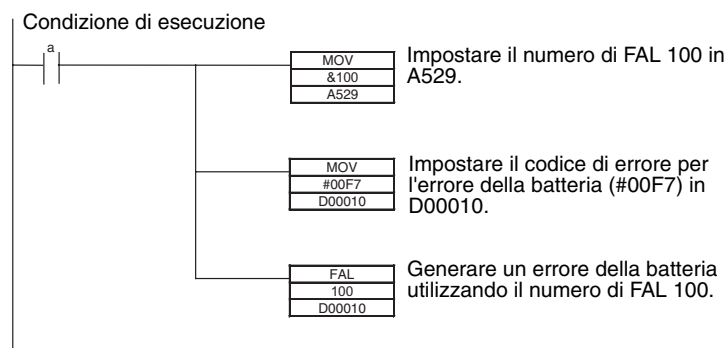
1,2,3...

1. Impostare in A529 il numero di FAL o FALS da utilizzare per la simulazione. A529 viene utilizzato per la simulazione degli errori tramite le istruzioni FAL(006) e FALS(007).
2. Impostare il numero di FAL o FALS da utilizzare per la simulazione come primo operando dell'istruzione FAL(006) o FALS(007).
3. Impostare il codice di errore e l'errore da simulare come secondo operando (due canali) di FAL(006) o FALS(007). Indicare un errore non fatale per FAL(006) e un errore fatale per FALS(007).

Per simulare più di un errore di sistema, utilizzare più istruzioni FAL(006) o FALS(007) come descritto.

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Funzionamento
Numero di FAL/FALS per la simulazione di errori di sistema	A529	Impostare un numero di FAL/FALS fittizio da utilizzare per simulare un errore di sistema. Da 0001 a 01FF esadecimale: numeri di FAL/ FALS da 1 a 511. Da 0000 o 0200 a FFFF esadecimale: nessun numero di FAL/FALS per la simulazione di errori di sistema.

Esempio per un errore della batteria

Nota Per cancellare gli errori di sistema simulati, utilizzare i metodi impiegati per gli errori effettivi. Per ulteriori informazioni, fare riferimento al manuale *CS-series Operation Manual* o al *Manuale dell'operatore della serie CJ*. È possibile cancellare tutti gli errori di sistema simulati con FAL(006) e FALS(007) riavviando il sistema.

6-7-6 Disabilitazione della memorizzazione degli errori FAL definiti dall'utente nel log degli errori

Questa funzione è supportata solo dalle CPU CS1-H, CJ1-H, CJ1M o CS1D.

Nella configurazione del PLC è disponibile un'impostazione che consente di impedire la registrazione di errori FAL definiti dall'utente, generati tramite l'istruzione FAL(006), e del monitoraggio del tempo per FPD(269) nel log degli

errori (da A100 a A199). L'errore FAL verrà comunque generato anche se si attiva questa impostazione, e verranno inoltre fornite le seguenti informazioni: flag di errore FAL (A40215), numeri di FAL eseguiti (da A360 a A391) e codice di errore (A400).

Questa funzione può essere utilizzata per memorizzare nel log degli errori solo gli errori FAL di sistema, escludendo gli errori definiti dall'utente generati dal programma tramite l'istruzione FAL(006), qualora troppo numerosi, che occuperebbero rapidamente tutto lo spazio disponibile del log.

Configurazione del PLC

Indirizzo della Console di programmazione		Nome	Impostazione	Valore predefinito	Frequenza di aggiornamento della CPU
Canale	Bit				
129	15	Impostazione di memorizzazione degli errori FAL definiti dall'utente	0: registrazione degli errori FAL definiti dall'utente nel log degli errori. 1: nessuna registrazione degli errori FAL definiti dall'utente nel log degli errori.	0: registrazione	A ogni esecuzione dell'istruzione FAL(006) (ogni ciclo)

Nota Anche se l'impostazione sopra descritta viene utilizzata per impedire la registrazione degli errori FAL definiti dall'utente, nel log degli errori vengono comunque memorizzati i seguenti elementi:

- Errori fatali definiti dall'utente (FALS(007))
- Errori di sistema non fatali
- Errori di sistema fatali
- Errori di sistema non fatali simulati dall'utente (FAL(006))
- Errori di sistema fatali simulati dall'utente (FALS(007))

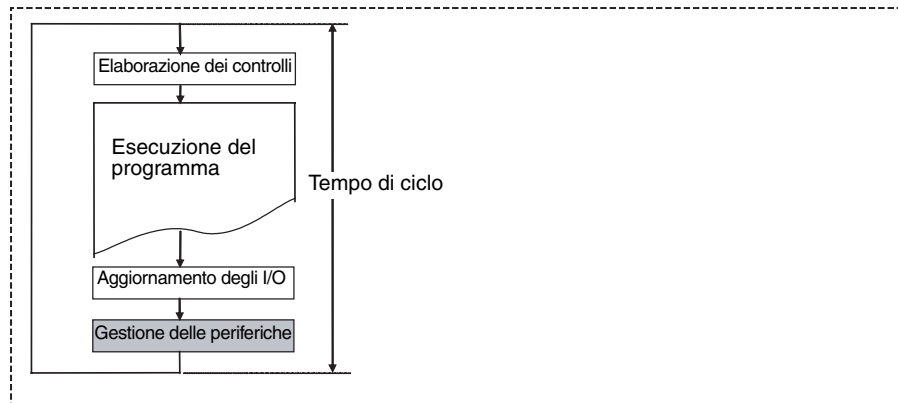
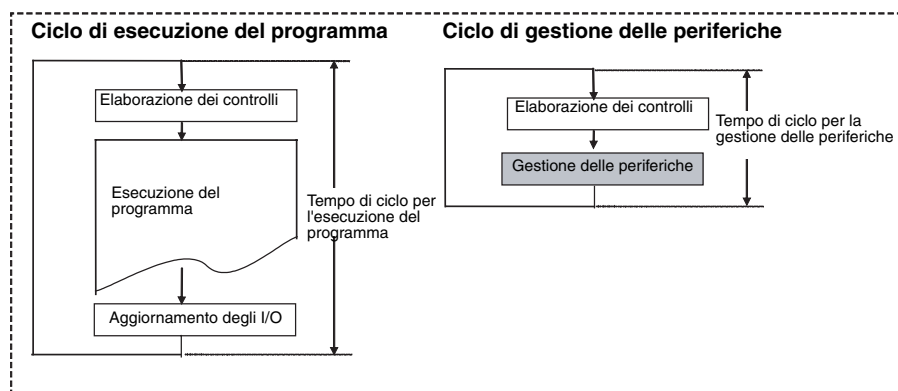
6-8 Modalità di elaborazione della CPU

6-8-1 Modalità di elaborazione della CPU

Di norma, la gestione delle periferiche (vedere nota) ha luogo al termine di ciascun ciclo, in seguito all'aggiornamento degli I/O, e ogni ciclo di gestione richiede il 4% del tempo di ciclo o un tempo impostato dall'utente. Ciò rende impossibile gestire i dispositivi periferici a una velocità superiore al tempo di ciclo e comporta un aumento del tempo di ciclo di un periodo pari al tempo necessario per la gestione delle periferiche.

Tuttavia, le CPU CS1D per sistemi a singola CPU oppure le CPU CS1-H o CJ1-H supportano modalità di elaborazione parallela che consentono di elaborare l'esecuzione del programma in parallelo con la gestione delle periferiche. Tali modalità consentono di ottenere tempi di gestione delle periferiche più rapidi e tempi di ciclo inferiori, in particolar modo se è richiesto un numero considerevole di operazioni di gestione. Le CPU CS1D per sistemi a due CPU e le CPU CJ1M non supportano le modalità di elaborazione parallela.

Nota La gestione delle periferiche include servizi non programmati richiesti dai dispositivi esterni, quali la gestione di eventi (ad esempio le comunicazioni per i comandi FINS) per Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS), nonché la gestione delle porte di comunicazione per le porte periferiche e RS-232C (esclusi i data link e altri aggiornamenti degli I/O speciali per i Moduli CPU bus).

Modalità normale

Modalità di elaborazione parallela

Modalità di elaborazione parallela

Esistono due diverse modalità di elaborazione parallela: l'elaborazione parallela con accesso alla memoria sincrono e l'elaborazione parallela con accesso alla memoria asincrono.

■ Elaborazione parallela con accesso alla memoria asincrono

In questa modalità, l'accesso alla memoria I/O per la gestione delle periferiche non è sincronizzato con l'accesso alla memoria I/O per l'esecuzione del programma. In altre parole, tutte le operazioni di gestione delle periferiche vengono eseguite in parallelo con l'esecuzione del programma, incluso l'accesso alla memoria. Questa modalità consente di ottenere i migliori tempi di esecuzione (rispetto alle altre modalità) sia per l'esecuzione del programma che per l'elaborazione di eventi qualora il carico di gestione delle periferiche sia particolarmente oneroso.

■ Elaborazione parallela con accesso alla memoria sincrono

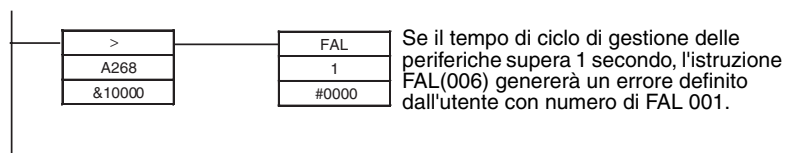
In questa modalità, l'accesso alla memoria I/O per la gestione delle periferiche non viene eseguito in parallelo all'esecuzione del programma bensì dopo l'esecuzione del programma, come accade nella modalità di esecuzione normale, vale a dire dopo il periodo di aggiornamento degli I/O. Tutte le operazioni di gestione delle periferiche vengono eseguite in parallelo con l'esecuzione del programma.

Questa modalità consente di ottenere tempi di esecuzione più rapidi rispetto alla modalità di esecuzione normale sia per quanto riguarda l'esecuzione del programma che l'elaborazione di eventi. Il tempo di ciclo relativo all'esecuzione del programma risulterà superiore rispetto alla modalità di elaborazione parallela con accesso alla memoria asincrono di un periodo pari al tempo necessario all'aggiornamento degli I/O per la gestione delle periferiche.

Nella tabella riportata di seguito sono indicati i tempi di ciclo e di risposta di gestione delle periferiche per la modalità normale, la modalità di elaborazione parallela con accesso alla memoria sincrono e la modalità di elaborazione parallela con accesso alla memoria asincrono. Tali valori si riferiscono a programmi costituiti da istruzioni di base con un tempo di ciclo di 10 ms e un Modulo Ethernet. I valori indicati vengono forniti a solo scopo di riferimento e possono variare a seconda del sistema.

Elemento	modalità normale	Elaborazione parallela con accesso alla memoria asincrono	Elaborazione parallela con accesso alla memoria sincrono
Tempo di ciclo	Impostato arbitrariamente su 1	0.9	0.9
Gestione delle periferiche	Impostato arbitrariamente su 1	0.4	1.0

Nota

1. La gestione delle periferiche include la gestione di eventi (ad esempio le comunicazioni per i comandi FINS) per Moduli di I/O speciale, Moduli CPU bus e schede interne (solo serie CS), nonché la gestione delle porte di comunicazione per le porte periferiche e RS-232C (esclusi i data link e altri aggiornamenti degli I/O speciali per i Moduli CPU bus).
2. Le CPU CS1 versione 1 o successiva e le CPU CS1-H o CJ1-H supportano anche una modalità di gestione prioritaria delle periferiche, che gestisce le periferiche a un ciclo prefissato durante l'esecuzione del programma consentendo di ottenere tempi di gestione più veloci rispetto alla modalità di elaborazione normale, sebbene l'esecuzione del programma risulti più lenta. La risposta degli eventi, tuttavia, non risulta altrettanto veloce come nelle modalità di elaborazione parallela. Si consiglia pertanto di utilizzare l'elaborazione parallela con accesso alla memoria asincrono quando la risposta agli eventi è l'elemento prioritario dell'elaborazione.
3. Quando si utilizza l'elaborazione parallela, è possibile nella CPU che si verifichino errori di superamento del tempo di ciclo, come descritto in *a)* e *b)* di seguito. Se si verificano i seguenti errori, il display del dispositivo di programmazione indicherà un tempo di ciclo eccessivamente lungo, verrà attivato il flag A40515 (flag di superamento del tempo di ciclo di gestione delle periferiche) e il funzionamento verrà interrotto (errore fatale):
   a) Se il tempo di ciclo di gestione delle periferiche supera 2,0 secondi, si verificherà un errore di superamento del tempo di ciclo. È possibile monitorare il tempo di ciclo di gestione delle periferiche in A268 per rilevare i possibili errori prima che si verifichino. È ad esempio possibile generare un errore definito dall'utente utilizzando il numero di FAL 001 nel caso in cui il tempo di ciclo di gestione delle periferiche superi 1 secondo, ovvero se il contenuto di A268 supera 2710 esadecimale (10000 decimale).

   b) È possibile che si verifichi un errore di superamento del tempo di ciclo di gestione delle periferiche anche quando il tempo di elaborazione per il ciclo di esecuzione dell'istruzione, vale a dire il tempo di esecuzione dell'istruzione, è troppo breve. Tale valore temporale viene memorizzato in A266 e A267 in modalità di esecuzione normale. Indicativamente, se il tempo di esecuzione dell'istruzione è pari o inferiore a 2 ms, si verificherà un errore di superamento del tempo di ciclo di gestione delle periferiche e

non sarà possibile utilizzare la modalità di elaborazione parallela. Durante il debug di singole sezioni del programma (che possono prevedere un tempo di esecuzione dell'istruzione molto breve), utilizzare la modalità normale per evitare il verificarsi di questo errore.

Quando le applicazioni utente vengono eseguite in modalità di elaborazione parallela, scollegare la Console di programmazione. Poiché alla Console di programmazione viene allocato un tempo di gestione per aumentare il tempo di risposta ai comandi inviati dai tasti della Console, ciò comporterebbe un aumento del tempo di gestione delle periferiche e la riduzione dell'efficacia dell'elaborazione parallela.

Configurazione del PLC

La modalità di elaborazione viene specificata nella configurazione del PLC.

Indirizzo della Console di programmazione		Nome	Impostazione	Valore predefinito	Frequenza di aggiornamento della CPU
Canale	Bit				
219	Da 08 a 15	Modalità di elaborazione della CPU	00 esadecimale: modalità normale 01 esadecimale: Elaborazione parallela con accesso alla memoria sincrono 02 esadecimale: Elaborazione parallela con accesso alla memoria asincrono Da 05 a FF esadecimale: frazionamento del tempo di esecuzione del programma per la modalità di gestione prioritaria delle periferiche (da 5 a 255 ms in incrementi di 1 ms) Le impostazioni 03 e 04 esadecimale non sono definite (illegali) e provocano errori di configurazione del PLC (non fatali).	00 esadecimale: modalità normale	Avvio dell'operazione

Flag e canali dell'area ausiliaria

Nome	Indirizzo	Funzionamento
Superamento del tempo di ciclo di gestione delle periferiche	A40515	Viene attivato quando il tempo di ciclo di gestione delle periferiche supera i 2 secondi. Il funzionamento viene interrotto.
Tempo di ciclo di gestione delle periferiche	A268	Contiene il tempo di ciclo di gestione delle periferiche quando viene utilizzata una delle modalità di elaborazione parallela (accesso alla memoria sincrono o asincrono) e il PLC è in modalità RUN o MONITOR. Il tempo è espresso in formato binario ed è compreso tra 0,0 e 2000,0 (in incrementi di 0,1 ms).
Tempo di esecuzione dell'istruzione (totale delle frazioni di tempo per l'esecuzione del programma e delle frazioni di tempo per la gestione delle periferiche)	A266 e A267	In modalità normale, è incluso solo il tempo di esecuzione dell'istruzione. Il tempo è memorizzato come valore binario a 32 bit. Da 00000000 a FFFFFFFF esadecimale (unità: 0,1 ms) (Da 0 a 429.496.729,5 ms) A266: canale meno significativo A267: canale più significativo

Elaborazione parallela con accesso alla memoria asincrono

Esecuzioni del programma

Controllo		Controllo bus di I/O e altre elaborazioni 0,3 ms
Tempo di esecuzione dell'istruzione		Tempo di esecuzione totale di tutte le istruzioni
Calcoli del tempo di ciclo minimo		Tempo di elaborazione per il tempo di ciclo minimo di esecuzione del programma
Gestione ciclica	Aggiornamento degli I/O	Tempo di aggiornamento degli I/O per ciascun Modulo x numero di Moduli
	Aggiornamento degli I/O speciali per Moduli CPU bus	Tempo di aggiornamento degli I/O speciali per ciascun Modulo x numero di Moduli
Gestione delle periferiche	Accesso ai file	Tempo di gestione delle periferiche impostato nella configurazione del PLC (predefinito: 4% del tempo di ciclo)

Gestione delle periferiche

Controllo		Controllo batteria, controllo memoria del programma utente etc. 0,2 ms
Gestione delle periferiche	Gestione eventi per Moduli di I/O speciale	Include la gestione degli eventi per l'accesso alla memoria I/O (vedere nota) Massimo 1 s per ciascun ciclo di gestione.
	Gestione degli eventi per i Moduli CPU bus	
	Gestione delle porte periferiche	
	Gestione delle porte RS-232C	
	Gestione eventi per schede interne (solo serie CS)	
	Gestione eventi per le porte di comunicazione (porte logiche interne) in uso (inclusa l'esecuzione in background)	

Nota La gestione degli eventi per l'accesso alla memoria I/O comprende 1) gestione dei comandi FINS ricevuti che accedono alla memoria I/O (comandi di lettura e scrittura della memoria I/O con codici comuni che iniziano con 01 esadecimale o comandi di impostazione e ripristino forzato con codici comuni che iniziano con 23 esadecimale) e 2) gestione dei comandi in modalità C che accedono alla memoria I/O (esclusi NT Link che utilizzano la porta periferiche o RS-232C).

Elaborazione parallela con accesso alla memoria sincrono

Esecuzioni del programma

Controllo		Controllo bus di I/O e altre elaborazioni 0,3 ms
Tempo di esecuzione dell'istruzione		Tempo di esecuzione totale di tutte le istruzioni
Calcoli del tempo di ciclo minimo		Tempo di elaborazione per il tempo di ciclo minimo di esecuzione del programma
Gestione ciclica	Aggiornamento degli I/O	Tempo di aggiornamento degli I/O per ciascun Modulo x numero di Moduli
	Aggiornamento degli I/O speciali per Moduli CPU bus	Tempo di aggiornamento degli I/O speciali per ciascun Modulo x numero di Moduli
Gestione delle periferiche	Accesso ai file	Tempo di gestione delle periferiche impostato nella configurazione del PLC (predefinito: 4% del tempo di ciclo)
	Gestione eventi che richiede l'accesso alla memoria I/O (vedere nota)	

Gestione delle periferiche

Controllo		Controllo batteria, controllo memoria del programma utente etc. 0,2 ms
Gestione delle periferiche	Gestione eventi per Moduli di I/O speciale	È esclusa la gestione degli eventi per l'accesso alla memoria I/O (vedere nota) Massimo 1 s per ciascun ciclo di gestione.
	Gestione degli eventi per i Moduli CPU bus	
	Gestione delle porte periferiche	
	Gestione delle porte RS-232C	
	Gestione eventi per schede interne (solo serie CS)	
	Gestione eventi per le porte di comunicazione (porte logiche interne) in uso (inclusa l'esecuzione in background)	

Nota La gestione degli eventi per l'accesso alla memoria I/O comprende 1) gestione dei comandi FINS ricevuti che accedono alla memoria I/O (comandi di lettura e scrittura della memoria I/O con codici comuni che iniziano con 01 esadecimale o comandi di impostazione e ripristino forzato con codici comuni che iniziano con 23 esadecimale) e 2) gestione dei comandi in modalità C che accedono alla memoria I/O (esclusi NT Link che utilizzano la porta periferiche o RS-232C).

6-8-2 Modalità di elaborazione parallela e tempi di ciclo minimi

Se, quando si utilizza una modalità di elaborazione parallela, viene specificato un tempo di ciclo minimo, dopo l'esecuzione del programma verrà inserita una pausa fino al raggiungimento del tempo di ciclo minimo, mentre la gestione delle periferiche continuerà normalmente.

6-8-3 Corrispondenza dei dati nell'elaborazione parallela con accesso alla memoria asincrono

Nei casi indicati di seguito i dati potrebbero non corrispondere durante l'utilizzo dell'elaborazione parallela con accesso alla memoria asincrono.

- Quando viene letto più di un canale dalla memoria I/O utilizzando un comando di comunicazione, i dati contenuti nei canali potrebbero non corrispondere.
- Se un'istruzione legge più di un canale della memoria I/O e durante l'esecuzione delle istruzioni viene eseguito il ciclo di gestione delle periferiche, i dati contenuti nei canali potrebbero non corrispondere.
- Se il medesimo canale della memoria I/O viene letto da più istruzioni in diverse posizioni del programma e tra l'esecuzione delle varie istruzioni viene eseguito il ciclo di gestione delle periferiche, i dati contenuti nei canali potrebbero non corrispondere.

Quando è necessario garantire la corrispondenza dei dati, è possibile eseguire le seguenti operazioni:

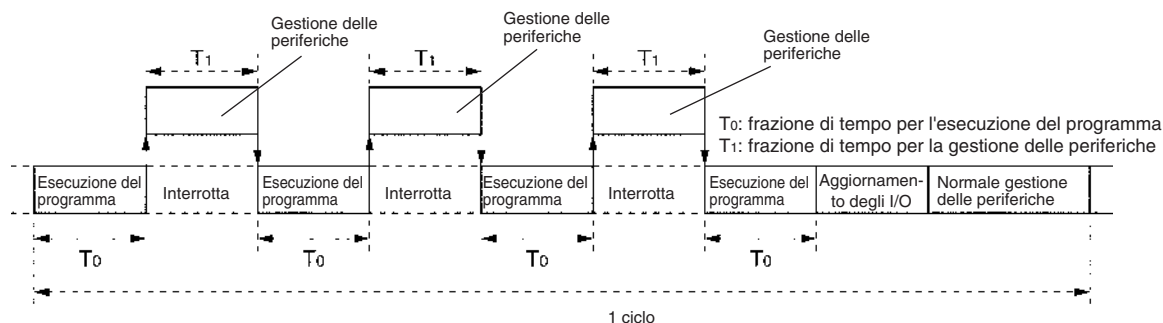
1. Utilizzare la modalità di elaborazione parallela con accesso alla memoria sincrono.
2. Utilizzare IOSP(287) per disabilitare la gestione delle periferiche, ove necessario all'interno del programma, quindi utilizzare IORS(288) per abilitarla nuovamente.

6-9 Modalità di gestione prioritaria delle periferiche

Le operazioni di gestione delle periferiche per la porta RS-232C, la porta periferiche, la scheda interna (solo serie CS), i Moduli CPU bus e i Moduli di I/O speciale vengono in genere eseguite una sola volta al termine del ciclo dopo l'aggiornamento degli I/O. A ciascun ciclo di gestione viene allocato il 4% del tempo di ciclo o un tempo impostato dall'utente. È tuttavia disponibile una modalità che consente di eseguire il ciclo di gestione periodicamente all'interno di un ciclo. Tale modalità, denominata modalità di gestione prioritaria delle periferiche, viene impostata nella configurazione del PLC.

Nota È possibile utilizzare la modalità di gestione prioritaria delle periferiche con le CPU della serie CJ o con le CPU della serie CS, ma per queste ultime il numero di lotto deve essere 001201□□□□ o successivo (data di produzione 1 dicembre 2000 o successiva). La modalità di gestione prioritaria delle periferiche non è supportata dalle CPU CS1D per sistemi a due CPU.

6-9-1 Modalità di gestione prioritaria delle periferiche

Quando si imposta la modalità di gestione prioritaria delle periferiche, l'esecuzione del programma viene interrotta nel momento specificato, vengono eseguite le operazioni di gestione necessarie, quindi l'esecuzione del programma riprende. Questa sequenza di azioni viene ripetuta durante l'esecuzione dell'intero programma. Al termine del periodo di aggiornamento degli I/O, viene inoltre eseguito il normale ciclo di gestione delle periferiche.

È pertanto possibile utilizzare la modalità di gestione prioritaria delle periferiche per gestire periodicamente determinate porte o Moduli, oltre al normale ciclo di gestione delle periferiche. Ciò consente di eseguire applicazioni che richiedono l'assegnazione della priorità alla gestione delle periferiche rispetto all'esecuzione del programma, quali le applicazioni di controllo dei processi in cui la velocità di risposta è essenziale per il monitoraggio host.

- È possibile specificare fino a cinque Moduli o porte per la gestione prioritaria. I Moduli CPU bus e i Moduli di I/O speciale delle serie CS e CJ vengono specificati mediante il numero di modulo.
- Durante ciascuna frazione di tempo allocata per la gestione prioritaria viene gestito un solo Modulo o una sola porta. Se il ciclo di gestione termina prima del tempo specificato, l'esecuzione del programma riprende immediatamente, e il ciclo di gestione della porta o del Modulo successivo non verrà eseguito fino alla frazione di tempo successiva prevista per la gestione prioritaria. È tuttavia possibile eseguire il ciclo di gestione per lo stesso Modulo o la stessa porta più di una volta all'interno di un ciclo.
- Le porte e i Moduli vengono gestiti nell'ordine in cui vengono rilevati dalla CPU.

Nota

1. Sebbene utilizzino le porte di comunicazione, le seguenti istruzioni vengono eseguite solo una volta durante il ciclo di esecuzione, anche quando si utilizza la modalità di gestione prioritaria delle periferiche:
   RXD(235) (RECEIVE)
   TXD(236) (TRANSMIT)
2. Se con un comando di comunicazione vengono letti più canali, quando si utilizza la modalità di gestione prioritaria delle periferiche non è possibile garantire la corrispondenza dei dati letti.
3. Quando si utilizza la modalità di gestione prioritaria delle periferiche, è possibile che la CPU superi il tempo di ciclo massimo. Il tempo di ciclo massimo è impostato nella configurazione del PLC come tempo di ciclo limite. Se il tempo di ciclo supera tale valore, verrà attivato il flag di superamento del tempo di ciclo massimo (A40108) e il funzionamento del PLC verrà interrotto. Quando si utilizza la modalità di gestione prioritaria delle periferiche, è necessario monitorare il tempo di ciclo corrente in A264 e A265 e adeguare il tempo di ciclo limite (indirizzo: +209) a seconda delle esigenze. La gamma di impostazione è compresa tra 10 e 40.000 ms, in incrementi di 10 ms, con un'impostazione predefinita di 1 secondo.

Impostazioni di configurazione del PLC

Per utilizzare la modalità di gestione prioritaria delle periferiche, è necessario effettuare le seguenti impostazioni nella configurazione del PLC:

- Frazionamento del tempo per l'esecuzione del programma: da 5 a 255 ms, in incrementi di 1 ms
- Frazionamento del tempo per la gestione delle periferiche: da 0,1 a 25,5 ms, in incrementi di 0,1 ms
- Moduli e/o porte per la gestione prioritaria: Modulo CPU bus (per numero di modulo)
Modulo di I/O speciale CS/CJ (per numero di modulo)
Scheda interna (solo serie CS)
Porta RS-232C
Porta periferiche

Indirizzo della Console di programmazione		Impostazioni	Valore predefinito	Funzione	Applicazione della nuova impostazione
Canale	Bit				
219	Da 08 a 15	00 Da 05 a FF (esadecimale)	00	00: disabilitazione della gestione prioritaria Da 05 a FF: frazionamento del tempo per l'esecuzione dell'istruzione (da 5 a 255 ms in incrementi di 1 ms)	Viene applicata all'avvio. Non è possibile modificarla durante il funzionamento.
	Da 00 a 07	Da 00 a FF (esadecimale)	00	00: disabilitazione della gestione prioritaria Da 01 a FF: frazionamento del tempo per la gestione delle periferiche: (da 0,1 a 25,5 ms in incrementi di 0,1 ms)	
220	Da 08 a 15	00 Da 10 a 1F Da 20 a 2F E1 FC FD (esadecimale)	00	00: disabilitazione della gestione prioritaria Da 10 a 1F: numero di modulo del Modulo CPU bus +10 (esadecimale) Da 20 a 7F: numero di modulo del Modulo di I/O speciale CS/CJ +20 (esadecimale) E1: scheda interna FC: porta RS-232C FD: porta periferiche	
	Da 00 a 07		00		
221	Da 08 a 15		00		
	Da 00 a 07		00		
222	Da 08 a 15		00		

- Funzionamento ed errori dipendono dalle impostazioni della configurazione del PLC, come indicato di seguito.
- Non è possibile effettuare alcuna impostazione da CX-Programmer per le CPU CS1 o CJ1. L'impostazione può essere effettuata da CX-Programmer versione 2.1 o successiva per le CPU CS1-H e CJ1-H.

Condizioni			Funzionamento della CPU	Errori di configurazione del PLC
Frazionamento del tempo per la gestione delle periferiche	Frazionamento del tempo per l'esecuzione dell'istruzione	Moduli e porte specificati		
Da 01 a FF: (da 0,1 a 25,5 ms)	Da 05 a FF: (da 5 a 255 ms)	Tutte le impostazioni corrette	Modalità di gestione prioritaria delle periferiche	Nessuno
		00 e impostazioni corrette		
		Impostazioni corrette ma ridondanti		
		Alcune impostazioni non corrette	Modalità di gestione prioritaria delle periferiche per le voci con impostazioni corrette	Generati
		Tutte le impostazioni 00	Funzionamento normale	Generati
		00 e impostazioni non corrette		
		Tutte impostazioni non corrette		
00	00	---	Funzionamento normale	Nessuno
Altro		---	Funzionamento normale	Generati

Nota Se viene rilevato un errore nella configurazione del PLC, il flag A40210 viene attivato e si verifica un errore non fatale.

Informazioni dell'area ausiliaria

Se si impostano le frazioni di tempo per l'esecuzione del programma e la gestione delle periferiche, il totale dei tempi di esecuzione del programma e dei tempi di gestione delle periferiche verrà memorizzato in A266 e A267. È possibile utilizzare tali informazioni per modificare nel modo più appropriato il frazionamento dei tempi.

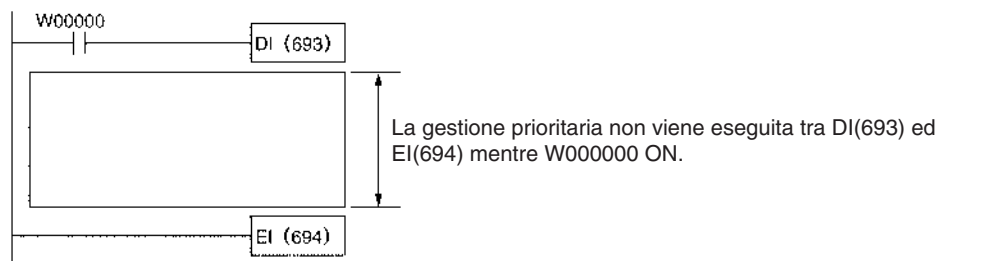
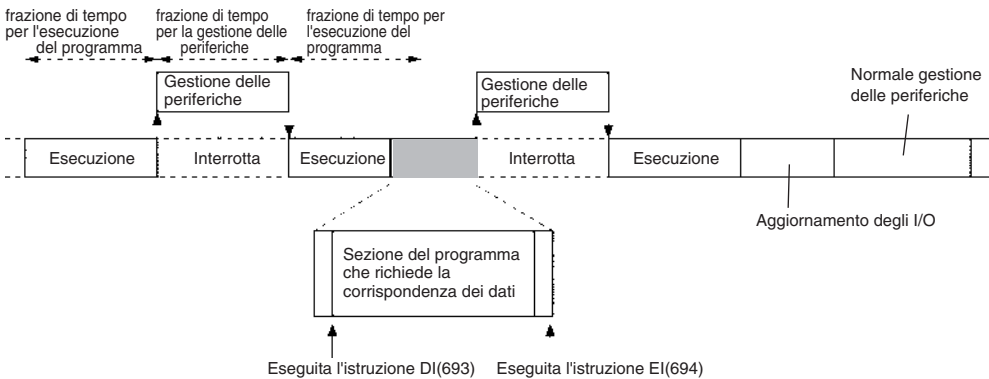
Se non si utilizza la modalità di gestione prioritaria delle periferiche, verrà memorizzato il tempo di esecuzione del programma. È possibile utilizzare tale valore per determinare le impostazioni più appropriate per il frazionamento dei tempi.

Canali	Contenuto	Descrizione	Aggiornamento
A266 e A267	Da 00000000 a FFFFFFFF esadecimale (da 0 a 4294967295 decimale)	Totale delle frazioni di tempo per l'esecuzione del programma e delle frazioni di tempo per la gestione delle periferiche Da 0,0 a 429.496.729,5 ms (incrementi di 0,1 ms) A267 (byte pi significativi) | A266 (byte meno significativi) | Il valore viene memorizzato come un numero binario a 32 bit (esadecimale a 8 cifre)	Il contenuto viene aggiornato a ogni ciclo e viene cancellato all'avvio.

6-9-2 Disabilitazione temporanea della modalità di gestione prioritaria delle periferiche

Quando si utilizza la modalità di gestione prioritaria delle periferiche, la corrispondenza dei dati non è garantita nei seguenti casi:

- Quando viene letto più di un canale da un dispositivo periferico tramite un comando di comunicazione. È possibile che i dati vengano letti durante diverse frazioni del ciclo di gestione delle periferiche, provocando la mancata corrispondenza dei dati.
- Quando nel programma si utilizzano istruzioni con tempi di esecuzione particolarmente lunghi, ad esempio per il trasferimento di grandi quantità di dati della memoria I/O. È possibile che l'operazione di trasferimento venga interrotta per l'esecuzione del ciclo di gestione delle periferiche, provocando la mancata corrispondenza dei dati. Ciò si verifica nel caso in cui i canali scritti dal programma vengano letti da una periferica prima che l'operazione di scrittura sia stata completata oppure quando i canali letti dal programma vengono scritti da una periferica prima che l'operazione di lettura sia stata completata.
- Quando due istruzioni accedono agli stessi canali in memoria. Se questi canali vengono scritti da un dispositivo periferico tra l'esecuzione delle due istruzioni, queste ultime leggeranno in memoria due valori diversi.

Qualora sia necessario garantire la corrispondenza dei dati, è possibile utilizzare le istruzioni DI(693) (DISABLE INTERRUPTS) e EI(694) (ENABLE INTERRUPTS) per le CPU CS1 o CJ1 in modo da impedire la gestione prioritaria nelle sezioni di programma in questione, come illustrato dall'esempio fornito di seguito. Per le CPU CS1D per sistemi a singola CPU e per le CPU CS1-H, CJ1-H o CJ1M, è possibile utilizzare le istruzioni DISABLE PERIPHERAL SERVICING e ENABLE PERIPHERAL SERVICING [IOSP(287) e IORS(288)].

Funzionamento

Nota

1. Le istruzioni DI(693) e IOSP(287) disabilitano non solo gli interrupt per la gestione prioritaria ma anche tutti gli altri interrupt, inclusi quelli di I/O, gli interrupt programmati ed esterni. Tutti gli interrupt generati verranno eseguiti dopo l'esecuzione del task ciclico (dopo l'esecuzione di END(001)), a meno che prima non venga eseguita l'istruzione CLI(691) per cancellare gli interrupt.
2. La disabilitazione degli interrupt tramite DI(693) o IOSP(287) rimane valida fino all'esecuzione di EI(694) o IORS(288), fino all'esecuzione di END(001) oppure fino all'arresto del PLC. Non è pertanto possibile creare sezioni di programma che si estendono oltre la fine di un task o ciclo. Utilizzare le istruzioni DI(693) e EI(694), o IOSP(287) e IORS(288), in ciascun task ciclico quando è necessario disabilitare gli interrupt in più cicli o task.

CPU CS1 e CJ1

Istruzione DI(693)

Quando viene eseguita l'istruzione DI(693), tutti gli interrupt vengono disabilitati, inclusi gli interrupt per la gestione prioritaria, gli interrupt di I/O, nonché gli interrupt programmati ed esterni, ad eccezione degli interrupt per i task relativi alle cadute di tensione. Se DI(693) viene eseguita quando gli interrupt sono già disabilitati, questi ultimi restano disabilitati.

Simbolo

Aree di programma applicabili

Area	Applicabilità
Aree di programmazione a blocchi	Sì
Aree di programmazione step	Sì
Programmi di subroutine	Sì
Task ad interrupt	No

Flag di condizione

Flag	Etichetta	Funzionamento
Flag di errore	ER	Viene attivato se DI(693) viene eseguita in un task ad interrupt. In caso contrario, rimane disattivato.

Istruzione EI(694)

Quando viene eseguita l'istruzione EI(694), tutti gli interrupt vengono abilitati, inclusi gli interrupt per la gestione prioritaria, gli interrupt di I/O, nonché gli interrupt programmati ed esterni, ad eccezione degli interrupt per i task relativi alle cadute di tensione. Se EI(694) viene eseguita quando gli interrupt sono già abilitati, questi ultimi restano abilitati.

Simbolo

Istruzione EI(694) — L'istruzione EI(694) non richiede alcuna condizione di esecuzione.

Aree di programma applicabili

Area	Applicabilità
Aree di programmazione a blocchi	Sì
Aree di programmazione step	Sì
Programmi di subroutine	Sì
Task ad interrupt	No

Flag di condizione

Flag	Etichetta	Funzionamento
Flag di errore	ER	Viene attivato se EI(694) viene eseguita in un task ad interrupt.

CPU CS1D per sistemi a singola CPU e CPU CS1-H, CJ1-H e CJ1M

Istruzione IOSP(287)

Quando viene eseguita l'istruzione IOSP(287), la gestione delle periferiche viene disabilitata. Se IOSP(287) viene eseguita quando la gestione delle periferiche è già disabilitata, quest'ultima rimane disabilitata.

Simbolo

Aree di programma applicabili

Area	Applicabilità
Aree di programmazione a blocchi	Sì
Aree di programmazione step	Sì
Programmi di subroutine	Sì
Task ad interrupt	No

Flag di condizione

Flag	Etichetta	Funzionamento
Flag di errore	ER	Viene attivato se IOSP(287) viene eseguita in un task ad interrupt. In caso contrario, rimane disattivato.

Istruzione IORS(288)

Quando viene eseguita l'istruzione IORS(288), la gestione delle periferiche precedentemente disabilitata tramite IOSP(287) viene nuovamente abilitata. Se IORS(288) viene eseguita quando la gestione delle periferiche è già abilitata, quest'ultima rimane abilitata.

Simbolo

Aree di programma applicabili

Area	Applicabilità
Aree di programmazione a blocchi	Sì
Aree di programmazione step	Sì
Programmi di subroutine	Sì
Task ad interrupt	No

Flag di condizione

Flag	Etichetta	Funzionamento
Flag di errore	ER	Viene attivato se IORS(288) viene eseguita in un task ad interrupt.

6-10 Funzionamento senza batteria

È possibile utilizzare i PLC delle serie CS e CJ senza installare una batteria, o nel caso in cui la batteria risulti scarica. La procedura utilizzata per il funzionamento senza batteria dipende dalle seguenti condizioni:

- CPU
- Se la memoria I/O (ad esempio l'area CIO) viene mantenuta o meno
- Se le aree DM ed EM vengono inizializzate all'avvio o meno
- Se le aree DM ed EM vengono inizializzate o meno dal programma utente

Nella seguente tabella sono sintetizzate le differenze sopra elencate.

<table>
<tr><th rowspan="3">CPU</th><th colspan="3">Nessun mantenimento della memoria I/O</th><th rowspan="3">Mantenimento della memoria I/O</th></tr>
<tr><th rowspan="2">Nessuna inizializzazione delle aree DM ed EM all'avvio</th><th colspan="2">Inizializzazione delle aree DM ed EM all'avvio</th></tr>
<tr><th>Da programma utente</th><th>Non da programma utente</th></tr>
<tr><td>CS1-H, CJ1-H, CJ1M o CS1D</td><td colspan="2">Utilizzare il funzionamento normale (con memoria flash) o una schedina di memoria.</td><td rowspan="2">Utilizzare il trasferimento automatico all'avvio da una schedina di memoria. Posizionare su ON il pin 2 del DIP switch.</td><td rowspan="2">Funzionamento non possibile con qualsiasi metodo.
È necessario installare una batteria.</td></tr>
<tr><td>CS1 o CJ1</td><td colspan="2">Utilizzare il trasferimento automatico all'avvio da una schedina di memoria. Posizionare su ON il pin 2 del DIP switch.</td></tr>
</table>

Nota

1. Quando si utilizza la CPU senza batteria, disabilitare la funzione di rilevamento di batteria scarica nella configurazione del PLC, indipendentemente dal metodo utilizzato per il funzionamento senza batteria.
2. Se la batteria non è installata o è scarica, il funzionamento della CPU sarà soggetto alle seguenti limitazioni, a prescindere dalla CPU in uso:
   - Lo stato del bit di disattivazione uscite (A50015) non sarà affidabile. Quando il bit di disattivazione uscite è impostato su ON, le uscite di tutti i Moduli di uscita vengono disattivate. Per impedire la disattivazione delle uscite di tutti i Moduli di uscita all'accensione, includere nel programma ladder le istruzioni riportate di seguito.

   - È possibile che il contenuto della memoria I/O, comprese le aree HR, DM ed EM, non venga mantenuto correttamente. Di conseguenza, impostare la configurazione del PLC in modo che lo stato del bit di ritenzione IOM (A50012) e del bit di ritenzione dello stato forzato (A50013) non vengano mantenuti all'accensione.
   - Non è possibile utilizzare le funzioni dell'orologio. I dati dell'orologio negli indirizzi da A351 a A354 e l'ora di avvio in A510 e A511 non saranno affidabili. Le date dei file scritti nella schedina di memoria dalla CPU non saranno ugualmente affidabili.
   - All'avvio i seguenti dati avranno tutti valore zero: tempo di esercizio (A523), ora caduta di tensione (A512 e A513) e numero di cadute di tensione (A514).
   - L'area del log degli errori da A100 a A199 non viene mantenuta.
   - Il numero di banco EM corrente sarà sempre 0 all'avvio.
   - All'avvio non sarà presente alcun file nella memoria file EM e non sarà possibile utilizzare le funzioni della memoria file. La memoria file EM

dovrà essere ripristinata nella configurazione del PLC e sarà necessario riformattarla.

CPU CS1-H, CJ1-H, CJ1M o CS1D

È possibile utilizzare le CPU CS1-H, CJ1-H, CJ1M o CS1D senza batteria per il funzionamento normale. Il programma utente e i dati dei parametri vengono automaticamente copiati nella memoria flash della CPU (backup automatico) e vengono automaticamente ripristinati dalla memoria flash all'avvio. In questo caso, la memoria I/O non viene mantenuta e sarà necessario inizializzare le aree DM ed EM dal programma utente.

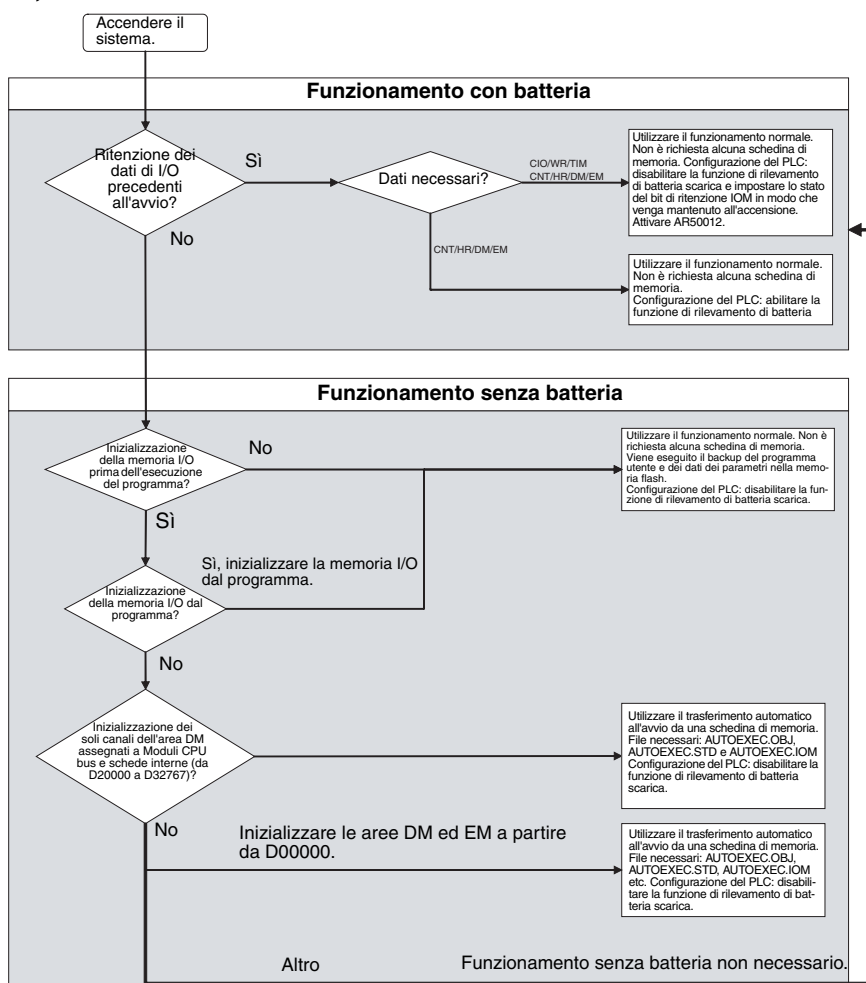
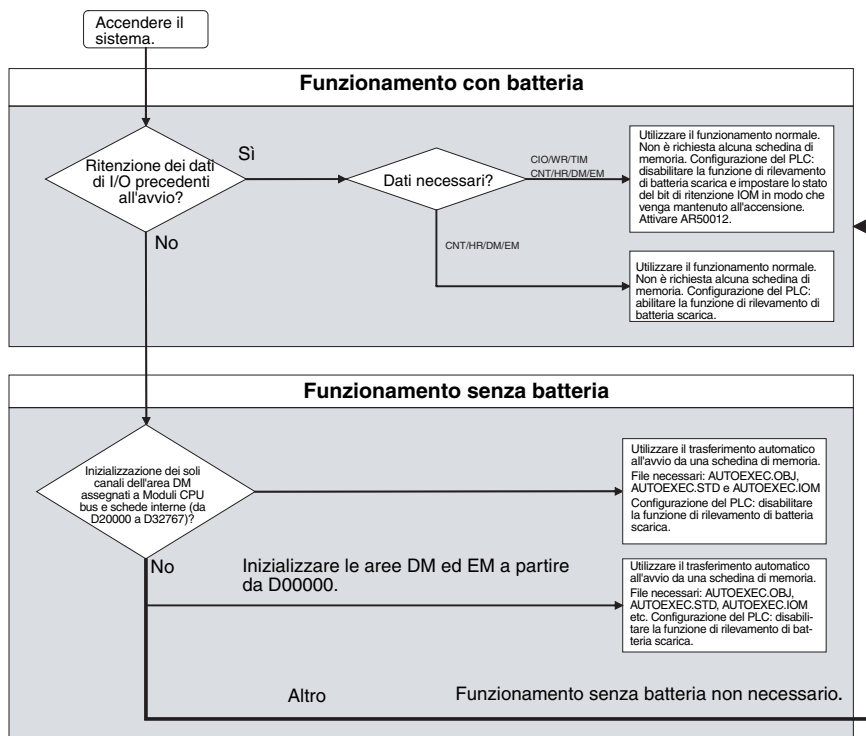
È altresì possibile utilizzare senza batteria anche le CPU CS1-H, CJ1-H, CJ1M o CS1D trasferendo automaticamente i dati da una schedina di memoria all'avvio, come per le CPU CS1. Nella schedina di memoria è possibile includere anche i dati delle aree DM ed EM.

CPU CS1 e CJ1

Le CPU CS1 e CJ1 possono essere utilizzate senza batteria trasferendo automaticamente i dati da una schedina di memoria all'avvio. In questo caso, i dati contenuti nella memoria I/O non verranno mantenuti. Nella schedina di memoria è possibile includere anche i dati delle aree DM ed EM.

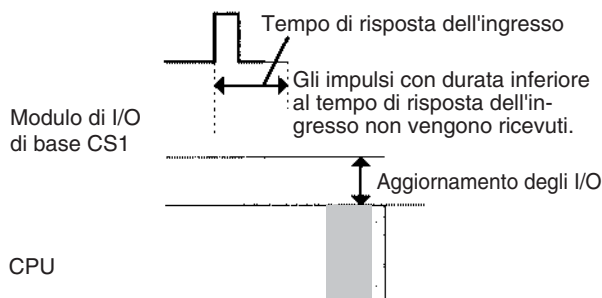
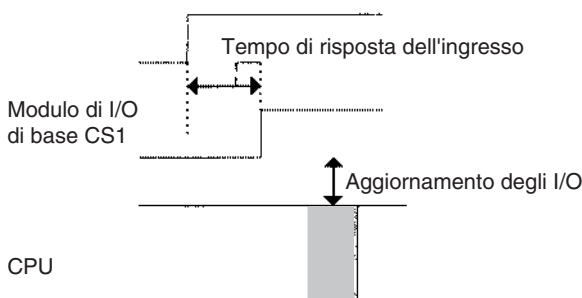
Procedura

I diagrammi di flusso riportati di seguito illustrano le procedure per i due tipi di CPU.

CPU CS1-H, CJ1-H, CJ1M o CS1D

CPU CS1 e CJ1

6-11 Altre funzioni

6-11-1 Impostazioni del tempo di risposta degli I/O

È possibile impostare i tempi di risposta di ingresso per i Moduli di I/O di base delle serie CS e CJ in base al numero di sistema e di slot. L'aumento del tempo di risposta di ingresso consente di ridurre gli effetti prodotti da irregolarità e disturbi. La riduzione del tempo di risposta di ingresso, mantenendo tuttavia un'ampiezza di impulso superiore al tempo di ciclo, consente di ricevere impulsi di ingresso più brevi.

Nota Con le CPU della serie CS è possibile inviare impulsi più brevi rispetto al tempo di ciclo utilizzando gli ingressi ad alta velocità disponibili in alcuni Moduli di I/O ad alta densità C200H o un Modulo di ingresso ad alta velocità. Per ulteriori informazioni, fare riferimento alla sezione *6-1-4 Ingressi ad alta velocità*.

Configurazione del PLC

È possibile impostare i tempi di risposta di ingresso per gli 80 slot di un PLC della serie CS o CJ (da Sistema 0, Slot 0 a Sistema 7, Slot 9) negli 80 byte degli indirizzi da 10 a 49.

Indirizzo della Console di programmazione	Nome	Impostazione (esadecimale)	Valore predefinito (esadecimale)
10 Bit da 0 a 7	Modulo di I/O di base CS/CJ Tempo di risposta di ingresso per Sistema 0, Slot 0	00: 8 ms 10: 0 ms 11: 0,5 ms 12: 1 ms 13: 2 ms 14: 4 ms 15: 8 ms 16: 16 ms 17: 32 ms	00 (8 ms)
:	:	:	:
49 Bit da 8 a 15	Modulo di I/O di base CS/CJ Tempo di risposta di ingresso per Sistema 7, Slot 9	Idem	00 (8 ms)

6-11-2 Assegnazione dell'area degli I/O

È possibile utilizzare un dispositivo di programmazione per impostare il primo canale per l'assegnazione degli I/O nei sistemi di espansione (sistemi di espansione della serie CS o CJ e sistemi di espansione I/O C200H). Questa funzione consente di fissare l'area di assegnazione degli I/O di ciascun sistema all'interno dell'intervallo da CIO 0000 a CIO 0999 (i primi canali vengono assegnati in base al numero di sistema).

CAPITOLO 7
Trasferimento del programma, esecuzione di prova e debug

In questo capitolo vengono descritti i processi utilizzati per il trasferimento del programma alla CPU e le funzioni per la verifica e il debug del programma.

7-1 Trasferimento del programma

È possibile utilizzare un dispositivo di programmazione per trasferire i programmi, la configurazione del PLC, i dati della memoria I/O e i commenti degli I/O alla CPU, quando la CPU è in modalità PROGRAM.

Procedura per il trasferimento del programma con CX-Programmer

1,2,3...

1. Fare clic su ***PLC, Transfer*** (PLC, Trasferisci), quindi su ***To PLC*** (Al PLC). Viene visualizzata la finestra di dialogo Download Options (Opzioni di scaricamento).
2. Specificare i dati da trasferire selezionando le opzioni appropriate: Programs (Programmi), Settings (Impostazioni, ossia la configurazione del PLC), I/O Table (Tabella di I/O), Symbols (Simboli), Comments (Commenti) e Program Index (Indice del programma).

   Nota È possibile selezionare la tabella di I/O e i commenti solo se sono presenti sulla schedina di memoria installata nella CPU.
3. Fare clic su **OK**.

È possibile trasferire il programma in uno dei due modi descritti di seguito.

- Trasferimento automatico all'accensione

Quando il sistema viene acceso, il file AUTOEXEC.OBJ nella schedina di memoria viene letto e trasferito nella CPU (il pin 2 del DIP switch deve essere impostato su ON).

- Sostituzione del programma durante il funzionamento

È possibile sostituire il file di programma esistente con il file di programma specificato nell'area ausiliaria attivando il bit di inizio sostituzione all'interno dell'area ausiliaria (A65015) direttamente dal programma durante il funzionamento della CPU. Per ulteriori informazioni, fare riferimento alla sezione *CAPITOLO 5 Funzioni della memoria file*.

Nota Se si utilizza CX-Programmer versione 4.0 o successiva con le CPU della serie CS/CJ versione 2.0 o successiva, i programmi di task possono essere scaricati singolarmente. Per ulteriori informazioni, fare riferimento a *1-4-1 Caricamento e scaricamento di singoli task* nel *Manuale dell'operatore della serie CJ*.

7-2 Esecuzione di prova e debug

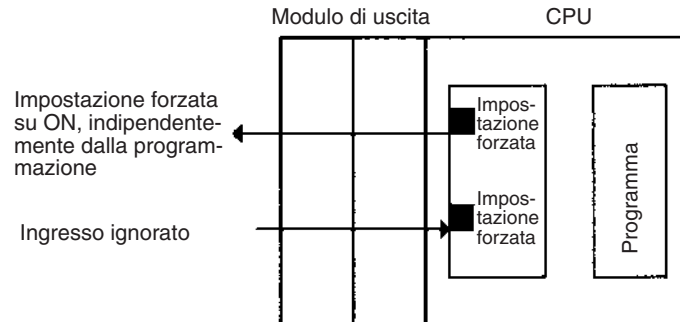
7-2-1 Impostazione e ripristino forzato

Mediante un dispositivo di programmazione è possibile impostare (ON) o ripristinare (OFF) in modo forzato i bit specificati, relativi all'area CIO, all'area ausiliaria, all'area HR e ai flag di completamento di temporizzatori e contatori. Lo stato forzato avrà priorità su quello determinato dal programma o dall'aggiornamento degli I/O. Tale stato non può essere sovrascritto per mezzo di istruzioni e, pertanto, verrà memorizzato a prescindere dallo stato del programma o da ingressi esterni finché non verrà cancellato tramite un dispositivo di programmazione.

Le operazioni di impostazione e ripristino forzato vengono utilizzate per forzare ingressi e uscite durante un'esecuzione di prova oppure per forzare alcune condizioni durante il debug.

Tali operazioni possono essere eseguite in modalità MONITOR o PROGRAM ma non in modalità RUN.

Nota Impostare contemporaneamente su ON sia il bit di ritenzione dello stato forzato (A50013) che il bit di ritenzione IOM (A50012) per conservare lo stato dei bit impostati o ripristinati in modo forzato durante il cambio di modalità operativa.

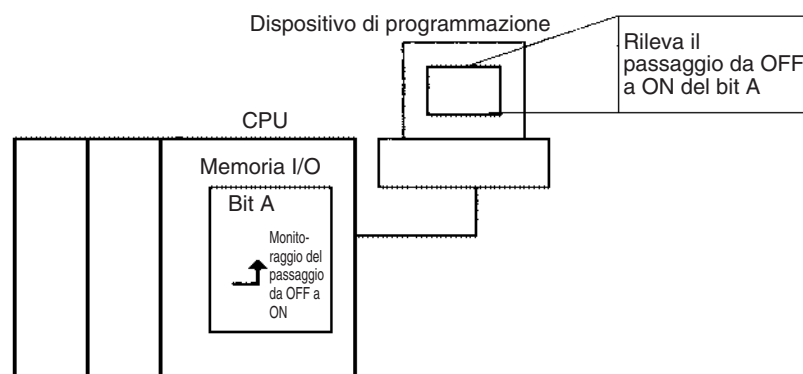
Attivare il bit di ritenzione dello stato forzato (A50013) e il bit di ritenzione IOM (A50012), quindi, nella configurazione del PLC, configurare l'impostazione Stato del bit di ritenzione dello stato forzato all'accensione in modo da mantenere attivato tale bit e garantire la conservazione dello stato dei bit impostati o ripristinati in modo forzato quando il sistema verrà spento.

È possibile impostare o ripristinare in modo forzato le seguenti aree.

CIO (bit di I/O, bit data link, bit del Modulo CPU bus, bit del Modulo di I/O speciale, bit della scheda interna, bit SYSMAC BUS, bit del Modulo di I/O ottico e bit di lavoro), area WR, flag di completamento dei temporizzatori, area HR, flag di completamento dei contatori. Le aree relative alla scheda interna, a SYSMAC BUS e ai terminali di I/O sono supportati solo dalle CPU della serie CS.

Utilizzo del dispositivo di programmazione

- Selezionare i bit per l'impostazione o il ripristino forzato.
- Selezionare l'impostazione o il ripristino forzato.
- Cancellare lo stato forzato (compresa la cancellazione contemporanea di tutti gli stati forzati).

7-2-2 Monitoraggio differenziale

Quando la CPU rileva che lo stato di un bit impostato mediante un dispositivo di programmazione è passato da OFF a ON o viceversa, i risultati vengono indicati nel flag di completamento del monitoraggio differenziale (A50809). Il flag viene attivato quando si verificano le condizioni impostate per il monitoraggio differenziale. Con un dispositivo di programmazione è possibile monitorare e visualizzare questi risultati su schermo.

Utilizzo del dispositivo di programmazione: CX-Programmer

1,2,3...

1. Fare clic con il pulsante destro del mouse sul bit da sottoporre al monitoraggio differenziale.
2. Scegliere ***Differential Monitor*** (Monitoraggio differenziale) dal menu PLC. Verrà visualizzata la finestra di dialogo per il monitoraggio differenziale.
3. Fare clic su ***Rising*** (Salita) oppure su ***Falling*** (Discesa).
4. Fare clic sul pulsante **Start** (Avvia). Quando viene rilevata la modifica specificata verrà emesso un segnale acustico e il conteggio verrà incrementato.
5. Fare clic sul pulsante **Stop** (Interrompi). Il monitoraggio differenziale verrà interrotto.

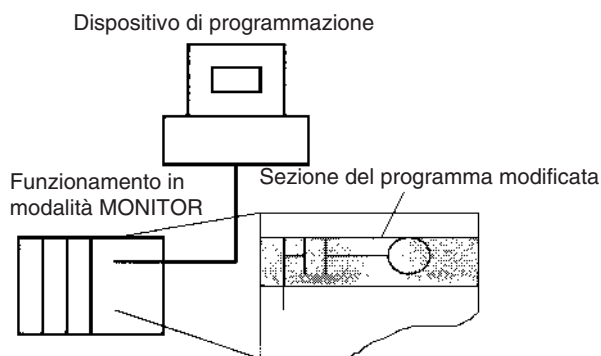
Bit e canali ausiliari correlati

Nome	Indirizzo	Descrizione
Flag di completamento del monitoraggio differenziale	A50809	Viene attivato quando, durante l'esecuzione del monitoraggio differenziale, si verifica la condizione corrispondente. Nota: il flag viene azzerato all'avvio del monitoraggio differenziale.

7-2-3 Modifica in linea

La funzione di modifica in linea viene utilizzata per aggiungere o modificare parte di un programma in una CPU direttamente dai dispositivi di programmazione quando la CPU è in modalità MONITOR o PROGRAM. Eventuali aggiunte o modifiche vengono apportate a un'istruzione alla volta se si utilizza la Console di programmazione e a una o più sezioni di programma alla volta con CX-Programmer. La funzione è quindi destinata a modifiche del programma di piccola entità senza interrompere il funzionamento della CPU.

La modifica in linea può essere eseguita contemporaneamente da più computer su cui è installato CX-Programmer oppure da una Console di programmazione, purché vengano modificati task diversi.

Modifica in linea

Se il programma nella CPU viene modificato in linea in modalità MONITOR, il tempo di ciclo aumenterà, con un'estensione pari a uno a più cicli.

Inoltre aumenterà il tempo di ciclo delle CPU CS1-H, CJ1-H, CJ1M e CS1D per il backup dei dati nella memoria flash dopo la modifica in linea. In questa fase l'indicatore BKUP sarà acceso. Lo stato di avanzamento del backup

viene visualizzato in CX-Programmer. Gli aumenti per ciascun ciclo sono riportati nella seguente tabella.

CPU	Aumento del tempo di ciclo	
	Modifica in linea	**Backup nella memoria flash**
CPU CS1 precedenti a EV1	90 ms max.	Non supportato
CPU CS1 EV1 o successive	12 ms max.	
CPU CS1-H		4% o tempo di ciclo
CPU CS1D		
CPU CS1		Non supportato
CPU CJ1-H		4% o tempo di ciclo
CPU CJ1M		

Per le CPU CS1-H, CJ1-H, CJ1M e CS1D esiste un limite al numero di modifiche che è possibile apportare consecutivamente. Il numero effettivo dipende dal tipo di modifica eseguita, tuttavia i valori riportati di seguito possono essere utilizzati come riferimento.

CJ1M-CPU□□: 40 modifiche
CS1G-CPU□□H/CJ1G-CPU□□H: 160 modifiche
CS1H-CPU□□H/CJ1H-CPU□□H/CS1D-CPU□□H/
CS1D-CPU□□S: 400 modifiche

Se si supera il limite, verrà visualizzato un messaggio in CX-Programmer o nella Console di programmazione e non sarà possibile eseguire ulteriori modifiche finché il backup dei dati della CPU non sarà completato.

Dimensione dei task ed estensione del tempo di ciclo

Di seguito è descritto il rapporto tra dimensione dei task sottoposti a modifica ed estensione del tempo di ciclo.

Quando si utilizza la versione 1 o successiva delle CPU CS1, CS1-H, CS1D, CJ1 o CJ1M, la dimensione del task (programma) modificato non ha praticamente alcun impatto sull'estensione del tempo di ciclo dovuta alla modifica in linea.

Quando si utilizza una CPU CS1 precedente a EV1, la dimensione del task modificato determina la durata dell'interruzione di un programma per la modifica in linea. Se si suddivide il programma in task di dimensioni minori, quando si utilizza la funzione di modifica in linea la durata dell'estensione del ciclo sarà inferiore rispetto ai modelli di PLC precedenti.

Precauzioni

Poiché il tempo di ciclo risulterà più lungo del normale quando un programma viene sovrascritto utilizzando la funzione di modifica in linea in modalità MONITOR, verificare che la durata dell'estensione non superi il tempo di ciclo di monitoraggio impostato nella configurazione del PLC. Se il tempo di monitoraggio viene superato, si verificherà un errore di superamento del tempo di ciclo e la CPU verrà arrestata. Riavviare la CPU selezionando la modalità PROGRAM prima di passare alla modalità RUN o MONITOR.

Nota Se il task modificato in linea contiene un programma a blocchi, i dati di esecuzione precedenti quali lo stato di attesa (WAIT) o pausa verranno cancellati dall'operazione di modifica in linea e l'esecuzione successiva partirà dall'inizio.

Modifica in linea da CX-Programmer

1,2,3...
1. Visualizzare la sezione del programma che verrà modificata.
2. Selezionare le istruzioni da modificare.
3. Fare clic su ***Program, Online Edit*** (Programma, Modifica in linea), quindi su ***Begin*** (Inizia).
4. Modificare le istruzioni.

5. Fare clic su ***Program, Online Edit*** (Programmi, Modifica in linea), quindi su ***Send Changes*** (Invia modifiche). Le istruzioni vengono controllate e, se non vi sono errori, trasferite alla CPU. A questo punto, le istruzioni nella CPU vengono sovrascritte e il tempo di ciclo risulta aumentato.

⚠ **Attenzione** Procedere con la modifica in linea solo dopo aver verificato che il tempo di ciclo esteso risultante non influisca sull'operazione. Se il tempo di ciclo è eccessivamente lungo, i segnali in ingresso potrebbero non essere recepiti.

Disabilitazione temporanea della modifica in linea

È possibile disabilitare la modifica in linea di un ciclo per verificare le caratteristiche di risposta per il controllo della macchina a livello del ciclo. La modifica in linea dal dispositivo di programmazione viene disabilitata per un ciclo e tutte le richieste di modifica in linea ricevute durante tale ciclo vengono sospese fino al ciclo successivo.

Per disabilitare la modifica in linea è sufficiente attivare il bit di disabilitazione della modifica in linea (A52709) e impostare su 5A la validità del bit di disabilitazione della modifica in linea (da A52700 ad A52707). Una volta eseguite queste impostazioni e ricevuta una richiesta di modifica in linea, l'operazione viene posta in attesa e viene attivato il flag di attesa per modifica in linea (A20110).

Quando si imposterà su OFF il bit di disabilitazione della modifica in linea (A52709), la modifica in linea verrà eseguita, il flag di elaborazione della modifica in linea (A20111) verrà attivato e il flag di attesa per modifica in linea (A20110) disattivato. Al termine della modifica in linea, il flag di elaborazione della modifica in linea (A20111) verrà disattivato.

È anche possibile disabilitare temporaneamente la modifica in linea attivando il bit di disabilitazione della modifica in linea (A52709) mentre l'operazione è in corso. Anche in questo caso il flag di attesa per modifica in linea (A20110) verrà attivato.

Se mentre la prima richiesta di modifica in linea è in attesa perviene una seconda richiesta, quest'ultima non verrà registrata e si verificherà un errore.

È possibile disabilitare questa funzione per evitare operazioni di modifica in linea accidentali. Come descritto in precedenza, disabilitare la modifica in linea attivando il bit di disabilitazione della modifica in linea (A52709) e impostando su 5A la validità del bit di disabilitazione della modifica in linea (da A52700 ad A52707).

Abilitazione della modifica in linea da un dispositivo di programmazione

Quando non è possibile abilitare la modifica in linea dal programma, è possibile utilizzare CX-Programmer.

1,2,3...

1. Esecuzione della modifica in linea con una Console di programmazione

   Se la modifica in linea viene eseguita da una Console di programmazione e lo stato di attesa dell'operazione non può essere cancellato, la Console di programmazione viene bloccata e non sarà possibile eseguire altre operazioni dalla Console.

   In tal caso, collegare CX-Programmer a un'altra porta seriale e impostare su OFF il bit di disabilitazione della modifica in linea (A52709). La modifica in linea verrà elaborata e sarà nuovamente possibile eseguire operazioni tramite la Console di programmazione.
2. Esecuzione della modifica in linea con CX-Programmer

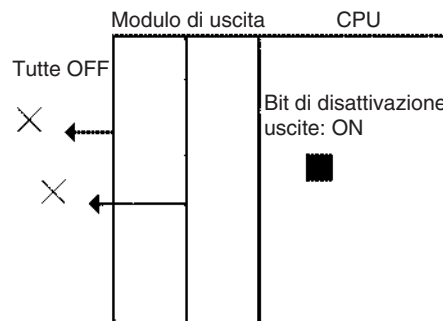
   Se la modifica in linea è in stato di attesa e le operazioni continuano, è possibile che CX-Programmer passi allo stato non in linea. In tal caso, ristabilire la connessione del computer al PLC e impostare su OFF il bit di disabilitazione della modifica in linea (A52709).

Bit e canali ausiliari correlati

Nome	Indirizzo	Descrizione
Validità del bit di disabilitazione della modifica in linea	Da A52700 ad A52707	Convalida il bit di disabilitazione della modifica in linea (A52709). Valore diverso da 5A: bit di disabilitazione della modifica in linea non valido 5A: bit di disabilitazione della modifica in linea valido
Bit di disabilitazione della modifica in linea	A52709	Per disabilitare la modifica in linea, attivare questo bit e impostare la validità del bit di disabilitazione della modifica in linea (da A52700 ad A52707) su 5A.
Flag di attesa per modifica in linea	A20110	Viene attivato quando un processo di modifica in linea è in attesa perché la funzione è stata disabilitata.
Flag di elaborazione della modifica in linea	A20111	Viene attivato quando è in esecuzione un processo di modifica in linea.

Disattivazione delle uscite

Se il bit di disattivazione uscite (A50015) viene impostato su ON mediante l'istruzione OUT o un dispositivo di programmazione, le uscite di tutti i Moduli di uscita verranno disattivate (comprese le uscite per uso generico o a impulsi integrate delle CPU CJ1M) e l'indicatore INH sulla parte frontale della CPU si accende.

Lo stato del bit di disattivazione uscite viene mantenuto anche quando il sistema viene spento e riacceso.

7-2-4 Registrazione dei dati

La funzione di registrazione dei dati consente di campionare i dati della memoria I/O specificati, utilizzando uno dei metodi indicati di seguito, e di memorizzarli nella memoria di registrazione, dove potranno essere letti e verificati in seguito tramite un dispositivo di programmazione.

- Tempo di campionamento specificato (da 10 a 2.550 ms in unità di 10 ms)
- Un campione per ciclo
- Esecuzione dell'istruzione TRACE MEMORY SAMPLING (TRSM)

È possibile specificare fino a 31 bit e 6 canali della memoria I/O per singolo campionamento. La memoria di registrazione ha una capacità di 4.000 canali.

Procedura di base

1,2,3...

1. Il campionamento viene avviato dopo l'impostazione dei parametri da CX-Programmer e quando viene eseguito il comando di inizio registrazione.
2. I dati campionati (dopo il punto 1 di cui sopra) vengono registrati quando si verifica la condizione di avvio della registrazione e i dati immediatamente successivi al ritardo (vedere nota 1) vengono memorizzati nella memoria di registrazione.
3. I dati della memoria di registrazione vengono campionati e la registrazione viene conclusa.

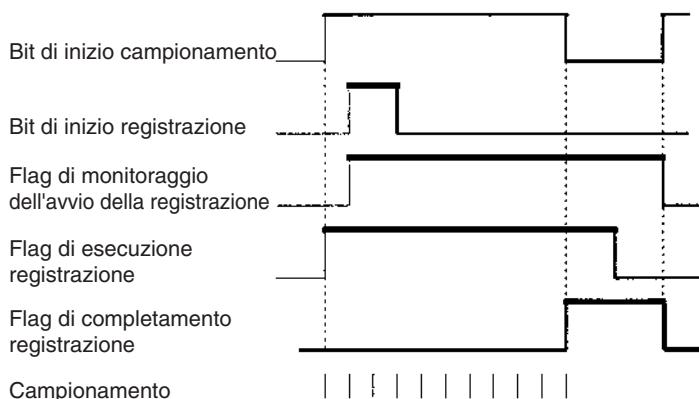
Nota Valore ritardo: indica il numero di periodi di campionamento per l'offset del campionamento nella memoria di registrazione a partire dall'attivazione del bit di inizio registrazione (A50814). Nella seguente tabella sono riportati gli intervalli di impostazione.

N. di canali campionati	Gamma di impostazione
0	Da –1999 a 2000
1	Da –1332 a 1333
2	Da –999 a 1000
3	Da –799 a 800
4	Da –665 a 666
5	Da –570 a 571
6	Da –499 a 500

Ritardo positivo: memorizzazione dei dati ritardata in base all'impostazione.

Ritardo negativo: memorizzazione dei dati precedenti in base al ritardo impostato.

Esempio un campionamento a 10 ms con un ritardo di –30 ms produce –30 x 10 = 300 ms, pertanto vengono memorizzati i dati di 300 ms prima dell'attivazione.

Nota Utilizzare un dispositivo di programmazione per attivare il bit di inizio campionamento (A50815). Non attivare mai il bit dal programma utente.

È possibile eseguire le registrazioni indicate di seguito.

Registrazione di dati programmata

La registrazione di dati programmata consente di campionare i dati a intervalli prefissati. I tempi di campionamento che è possibile specificare vanno da 10 a 2.550 ms in unità di 10 ms. Non utilizzare l'istruzione TRSM nel programma utente e accertarsi di aver impostato un periodo di campionamento superiore a 0.

Registrazione dei dati di un ciclo

La registrazione dei dati di un ciclo consente di campionare i dati dell'aggiornamento degli I/O al termine dell'esecuzione dei task dell'intero ciclo. Non utilizzare l'istruzione TRSM nel programma utente e accertarsi di aver impostato un periodo di campionamento superiore a 0.

Registrazione di dati tramite istruzione TRSM

Viene eseguito un solo campionamento all'esecuzione dell'istruzione TRACE MEMORY SAMPLING (TRSM). Quando nel programma si utilizza più di un'istruzione TRSM, viene eseguito un campionamento a ogni esecuzione dell'istruzione TRSM dopo il verificarsi della condizione di inizio della registrazione.

Procedura di registrazione dei dati

Per eseguire una registrazione, seguire la procedura descritta.

1,2,3...
1. Utilizzare CX-Programmer per impostare i parametri di registrazione (eseguire *PLC/Data Trace (PLC/Registrazione dati)* ed effettuare le impostazioni in *Execute/Set (Esegui/Imposta)*): indirizzo dei dati campionati, periodo di campionamento, ritardo e condizioni di attivazione.
2. Utilizzare CX-Programmer per avviare il campionamento o attivare il bit di inizio campionamento (A50815).
3. Ottenere la condizione di attivazione della registrazione.
4. Terminare la registrazione.
5. Utilizzare CX-Programmer per leggere i dati registrati.
   a) Scegliere ***Data Trace*** (Registrazione dati) dal menu PLC.
   b) Scegliere ***Select*** (Seleziona) dal menu Execution (Esecuzione).
   c) Scegliere ***Execute*** (Esegui) dal menu Execution (Esecuzione).
   d) Scegliere ***Read*** (Leggi) dal menu Execution (Esecuzione).

Bit e canali ausiliari correlati

Nome	Indirizzo	Descrizione
Bit di inizio campionamento	A50815	Utilizzare un dispositivo di programmazione per attivare il bit di inizio campionamento. È necessario che il bit venga attivato mediante un dispositivo di programmazione. Non attivare e disattivare il bit dal programma utente. Nota: al termine della registrazione dei dati il bit viene azzerato.
Bit di inizio registrazione	A50814	Quando lo stato del bit è ON, l'attivazione della registrazione viene monitorata e, ogni volta che si verificano le condizioni di attivazione della registrazione, i dati campionati vengono memorizzati nella memoria di registrazione. Questo bit consente di attivare le registrazioni riportate di seguito. 1) Registrazione programmata (registrazione a intervalli prefissati compresi tra 10 e 2.550 ms) 2) Registrazione tramite istruzione TRSM (registrazione all'esecuzione di TRSM) 3) Registrazione di un ciclo (registrazione al termine dell'esecuzione di tutti i task ciclici)
Flag di monitoraggio dell'avvio della registrazione	A50811	Questo flag viene attivato quando si verifica la condizione di attivazione della registrazione in seguito all'attivazione del bit di inizio registrazione. Il flag viene disattivato quando il campionamento viene nuovamente avviato attivando il bit di inizio campionamento.
Flag di esecuzione registrazione	A50813	Questo flag viene attivato quando il campionamento viene avviato da un bit di inizio campionamento, e viene disattivato al termine della registrazione.
Flag di completamento registrazione	A50812	Questo flag viene attivato quando la memoria di registrazione risulta piena, una volta verificatasi la condizione di attivazione della registrazione durante un'operazione di registrazione, e viene disattivato all'avvio dell'operazione di campionamento successiva.

Appendice A

Tavole di confronto dei PLC: PLC serie CJ, serie CS, C200HG/HE/HX, CQM1H, CVM1 e serie CV

Confronto delle caratteristiche funzionali

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Caratteristiche di base	Capacità	N. punti di I/O	2.560 punti	5.120 punti	1.184 punti	6.144 punti	512 punti
		Capacità di programma	250.000 step Uno step equivale in genere a un word. Per ulteriori informazioni, fare riferimento alla parte finale del capitolo *10-5 Tempi di esecuzione delle istruzioni e numero di step* nel *Manuale dell'operatore*.	250.000 step Uno step equivale in genere a una word. Per ulteriori informazioni, fare riferimento alla parte finale del capitolo *10-5 Tempi di esecuzione delle istruzioni e numero di step* nel *Manuale dell'operatore*.	2.000 word (63.200 word per -Z)	62.000 word	15.200 word
		Memoria dati massima	32.000 word	32.000 word	6.000 word	24.000 word	6.000 word
		Bit di I/O	160 word (2.560 bit)	320 word (5.120 bit)	40 word (640 bit)	128 word (2.048 bit)	32 word (512 bit)
		Bit di lavoro	2.644 word (42.304 bit) + WR: 512 word (8.192 bit) = 3.156 word (50.496 bit)	2.644 word (42.304 bit) + WR: 512 word (8.192 bit) = 3.156 word (50.496 bit)	408 word (6.528 bit)	168 word (2.688 bit) +400 word (6.400 bit)	158 word (2.528 bit)
		Bit di ritenzione	512 word (8.192 bit)	512 word (8.192 bit)	100 word (1.600 bit)	300 word (4.800 bit) Max.: 1.400 word (2, 400 bit)	100 word (1.600 bit)
		Memoria dati estesa massima	32.000 word x 13 banchi	32.000 word x 13 banchi	6.000 wordx 3 banchi (6.000 word x 16 banchi per -Z)	32.000 word x 8 banchi (opzionale)	6.000 word
		N. massimo temporizzatori/contatori	4.096 ciascuno	4.096 ciascuno	Temporizzatori/contatori combinati: 512	1.024 punti	Temporizzatori/contatori combinati: 512
	Velocità di elaborazione	Istruzioni di base (LD)	CJ1: 0,08 µs minimo CJ1-H: 0,02 µs minimo CJ1M: 0,1 µs minimo	CS1: 0,04 µs minimo CS1-H: 0,02 µs minimo	0,104 µs minimo	0,125 µs minimo	0,375 µs minimo
		Istruzioni speciali (MOV)	CJ1: 0,25 µs minimo CJ1-H: 0,18 µs minimo CJ1M: 0,3 µs minimo	CS1: 0,25 µs minimo CS1-H: 0,18 µs minimo	0,417 µs minimo	4,3 µs minimo	17,7µs
		Tempo di esecuzione dei processi comuni del sistema	CJ1: 0,5 ms minimo CJ1-H: 0,3 ms minimo in modalità normale; 0,2 ms in modalità di elaborazione parallela CJ1M: 0,5 ms minimo	CS1: 0,5 ms minimo CS1-H: 0,3 ms minimo in modalità normale; 0,2 ms in modalità di elaborazione parallela	0,7 ms	0,5 ms	0,7 ms
		Ritardo durante la modifica in linea (scrittura)	CJ1: circa 12 ms CJ1-H: circa 11 ms per CPU4□ e 8 ms per CPU6 CJ1M: circa 14 ms	CS1: circa 12 ms CS1-H: circa 11 ms per CPU4□ e 8 ms per CPU6	80 ms (160 ms per -Z)	500 ms	Tipicamente 250 ms
Struttura	Montaggio con viti		No	Sì	Sì	Sì	No
	Montaggio su guida DIN		Sì	Sì	Sì	No	Sì
	Rack		No	Sì	Sì	Sì	No
	Dimensioni (A x P, mm)		90 x 65	130 x 123	130 x 118	250 x 100	110 x 107
Numero di Moduli/sistemi	Moduli di I/O		40 Moduli	89 Moduli (sistemi slave inclusi)	10 o 16 Moduli	64 Moduli (8 sistemi x 8 Moduli)	16 Moduli
	Moduli CPU bus		16 Moduli	16 Moduli	Nessuno	16 Moduli	Nessuno
	Sistemi di espansione I/O		3 sistemi	7 sistemi	3 sistemi	7 sistemi	1 sistema

Elemento		Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Funzioni di task		Sì	Sì	No	No	No
Modalità di elaborazione della CPU (esecuzione programma e gestione periferiche)	Modalità normale	Sì	Sì	---	---	---
	Modalità di gestione prioritaria delle periferiche	Sì	Sì	---	---	---
	Elaborazione parallela con accesso alla memoria sincrono	CJ1: No CS1-H: Sì CJ1M: No	CS1: No CS1-H: Sì	No	No	No
	Elaborazione parallela con accesso alla memoria asincrono	CS1: No CJ1-H: Sì CJ1M: No	CS1: No CS1-H: Sì	No	No	No
Formato aggiornamento degli I/O	Aggiornamento ciclico	Sì	Sì	Sì	Sì	Sì
	Aggiornamento programmato	No	No	No	Sì	No
	Aggiornamento con passaggio a zero	No	No	No	Sì	No
	Aggiornamento immediato	Sì	Sì	No	Sì	No
	Aggiornamento immediato tramite istruzione IORF	Sì	Sì	Sì	Sì	Sì
Funzioni di orologio		Sì	Sì	Sì	Sì	Sì (è richiesta una cartuccia di memoria)
uscita RUN		Sì (a seconda del Modulo di alimentazione)	Sì (a seconda del Modulo di alimentazione)	Sì (a seconda del Modulo di alimentazione)	Sì	No
Modalità di avvio (impostazione predefinita della configurazione del PLC quando non è collegata una Console di programmazione)		modalità RUN	CS1: Modalità PROGRAM CS1-H: modalità RUN	modalità RUN	modalità RUN	Modalità PROGRAM
Disabilitazione elaborazione dell'interrupt di spegnimento		CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
Funzionamento senza batteria		CJ1: schedina di memoria CJ1-H: schedina di memoria o memoria flash CJ1M: schedina di memoria o memoria flash	CS1: schedina di memoria CS1-H: schedina di memoria o memoria flash	Schedina di memoria	Schedina di memoria	Cartuccia di memoria
Backup automatico in memoria flash		CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
Continuazione al riavvio		No	No	No	Sì	No
Memoria esterna	Supporto	Schedina di memoria (ROM flash)	Schedina di memoria (ROM flash)	Cartuccia di memoria (EEPROM, EPROM)	Schedina di memoria (RAM, EEPROM, EPROM)	Cartuccia di memoria (ROM, EEPROM, EPROM)
	Capacità	48 MB	48 MB	4 a 32.000 word (da 4.000 a 64.000 word per -Z)	Da 32.000 a 512.000 word (RAM: da 64 a 512 KB; EEPROM: da 64 a 128 KB; EPROM: da 0,5 a 1 MB)	Da 4.000 a 16.000 word
	Contenuto	Programmi, memoria I/O, parametri	Programmi, memoria I/O, parametri	Programmi, memoria I/O, parametri	Programmi, memoria I/O, parametri	Programmi, DM di sola lettura, parametri
	Metodo di lettura/scrittura	Dispositivo di programmazione, programma utente (istruzioni per la memoria file) o Host Link	Dispositivo di programmazione, programma utente (istruzioni per la memoria file) o Host Link	Attivazione del bit SR	Dispositivo di programmazione, programma utente (istruzioni per la memoria file), Host Link o dispositivo di scrittura della schedina di memoria	Attivazione del bit AR
	Formato dei file	Binario	Binario	Binario	Binario	Binario
	Memoria dati estesa gestita come file	Sì (ad eccezione delle CPU CJ1M)	Sì	No	No	No
	Programmi trasferiti automaticamente all'avvio	Sì	Sì	Sì	Sì	Sì
Scheda interna		No	Scheda di comunicazione seriale	Scheda di comunicazione	No	Scheda di comunicazione
Porte seriali integrate		Sì (RS-232C x 1)	Sì (RS-232C x 1)	Sì (RS-232C x 1)	Sì (RS-232C o RS-422 x 1)	Sì (RS-232C x 1)

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Comunicazione seriale	Porta periferiche	Bus di periferica	Sì	Sì	Sì	Sì	Sì
		Host Link (SYSMAC WAY)	Sì	Sì	Sì	No (possibile con collegamento a un'interfaccia di periferica)	Sì
		Gateway seriale (conversione in CompoWay/F)	Sì (CPU versione 3.0 o successiva)	Sì (CPU versione 3.0 o successiva)	No	No	No
		Senza protocollo	No	No	Sì	No	Sì
		NT Link	Sì	Sì	No	No	No
	Porta RS-232C integrata nella CPU	Bus di periferica	Sì	Sì	Sì	No	No
		Host Link (SYSMAC WAY)	Sì	Sì	Sì	Sì	Sì
		Gateway seriale (conversione in CompoWay/F)	Sì (CPU versione 3.0 o successiva)	Sì (CPU versione 3.0 o successiva)	No	No	No
		Senza protocollo	Sì	Sì	Sì	No	Sì
		NT Link	Sì (1:N)	Sì (1:N)	Sì	No	Sì (1:1)
		collegamenti seriali tra PLC	Sì (solo CJ1M)	No	No	No	No
	Porta RS-232C o RS-422/RS-485 sulla scheda di comunicazione	Bus di periferica	No	No	Sì	No	No
		Host Link (SYSMAC WAY)	No	Sì I comandi WG, MP e CR non sono supportati.	Sì Il comando CR non è supportato.	Sì I comandi WG e MP non sono supportati.	Sì Il comando CR non è supportato.
		Gateway seriale (conversione in CompoWay/F, Modbus-RTU, Modbus-ASCII o Host Link FINS)	No	Sì (scheda/modulo di comunicazione seriale versione 1.2 o successiva)	No	No	No
		Senza protocollo	No	No	Sì	No	Sì
		NT Link	No	Sì	Sì	No	Sì (1:1 e 1:N)
		Protocol macro	No	Sì	Sì	No	Sì
		Master CompoWay/F	No	Sì (utilizzando protocol macro)	Sì (utilizzando protocol macro)	No	Sì (utilizzando protocol macro)

Elemento		Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Interrupt	Interrupt di I/O	Sì (massimo 2 Moduli di interrupt di ingresso: 32 punti più 4 punti per gli I/O integrati delle CPU CJIM). Gli interrupt di I/O non sono supportati dalle CPU CJ1.	Sì (massimo 4 o 2 Moduli di interrupt di ingresso: 32 punti)	Sì (massimo 2 Moduli di interrupt di ingresso: 16 punti)	Sì (massimo 4 Moduli di interrupt di ingresso: 32 punti)	Sì (4 integrati nel Modulo CPU bus)
	Interrupt programmati	Sì	Sì	Sì	Sì	Sì
	Interrupt di temporizzatori a singola esecuzione	No	No	No	No	Sì
	Interrupt di ingresso in modalità contatore	Sì (solo CPU CJ1M)	No	No	No	Sì
	Interrupt di contatore veloce	Sì (solo CPU CJ1M)	No	No	No	Sì
	Interrupt esterni	Sì (le CPU CJ1 non supportano interrupt esterni)	Sì	No	No	No
	Dalla scheda di comunicazione	No	Sì	Sì	No	No
	Interrupt di accensione	No	No	No	Sì	No
	Interrupt di spegnimento	Sì	Sì	No	Sì	No
	Tempo di risposta degli interrupt	0,17 ms I/O integrati delle CPU CJ1M: 0,12 ms	Modulo di I/O speciale C200H: 1 ms Moduli di I/O della serie CJ: 0,1 ms	1 ms	---	Circa 0,1 ms
Area di configurazione del PLC		Nessun indirizzo utente (impostazione possibile solo dal dispositivo di programmazione, inclusa la Console di programmazione)	Nessun indirizzo utente (impostazione possibile solo dal dispositivo di programmazione, inclusa la Console di programmazione)	Assegnazione fissa dell'area DM: da DM 6600 a DM 6655, da DM 6550 a DM 6559 (impostazione possibile dalla Console di programmazione)	Nessun indirizzo utente (impostazione possibile solo dal dispositivo di programmazione e in parte dalla Console di programmazione)	Assegnazione fissa dell'area DM: da DM 6600 a DM 6655 (impostazione possibile dalla Console di programmazione)

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Impo-stazioni iniziali	I/O	Tempo di risposta di ingresso del Modulo di I/O di base	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	No	No	Impostata nella configurazione del PLC
		Primi indirizzi dei sistemi	Impostati nella tabella di I/O dal dispositivo di programmazione (l'ordine dei numeri di sistema è fisso)	Impostati nella tabella di I/O dal dispositivo di programmazione (l'ordine dei numeri di sistema è fisso)	No	Impostati nella configurazione del PLC (è possibile impostare l'ordine dei numeri di sistema)	No
		Primo indirizzo dei Moduli di I/O ottici SYSMAC BUS dal master	No	No	No	Impostata nella configurazione del PLC	No
		Funzionamento in seguito a errore di verifica degli I/O	No	No	No	Impostata nella configurazione del PLC	No
	Memo-ria	Protezione della memoria utente	Impostata tramite DIP switch	Impostata tramite DIP switch	Impostata tramite DIP switch	Determinata dall'impostazione degli interruttori associati ai tasti	Impostata tramite DIP switch
		Aree di ritenzione	No	No	No	Impostata nella configurazione del PLC	No
		Ritenzione delle word di I/O in seguito a errori fatali (tranne caduta di tensione)	No	No	No	Impostata nella configurazione del PLC	No
		Salvataggio memoria mediante bit di ritenzione IOM all'accensione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC
		Salvataggio memoria mediante bit di ritenzione dello stato forzato all'accensione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC
		Monitoraggio stato del DIP switch	Sì	Sì	Sì	No	Sì
	Istru-zioni	Impostazione dati DM indiretti in formato BCD o in formato binario	Immissione diretta possibile	Immissione diretta possibile	No	Impostata nella configurazione del PLC	No
		Utilizzo di più istruzioni JMP(0)	Utilizzo di più istruzioni già possibile	Utilizzo di più istruzioni già possibile	No	Impostata nella configurazione del PLC	No
		Funzionamento in seguito a errori di istruzione (continuazione o interruzione)	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	No	No	No
		Esecuzione in background	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
	Memo-ria file	Trasferimento automatico all'avvio	Determinato dall'impostazione del DIP switch (lettura automatica dalla schedina di memoria)	Determinato dall'impostazione del DIP switch (lettura automatica dalla schedina di memoria)	Determinato dall'impostazione del DIP switch (lettura automatica dalla cartuccia di memoria)	Impostato nella configurazione del PLC o determinato dall'impostazione del DIP switch (lettura automatica dalla schedina di memoria)	Determinato dall'impostazione del DIP switch (lettura automatica dalla schedina di memoria)
		Conversione in file nell'area EM	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	No	No	No
	Interrupt	Risposta degli interrupt	No	No	Impostata nella configurazione del PLC (C200H/risposta ad alta velocità)	No	No
		Rilevamento degli errori	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	No	No
		Ritenzione degli interrupt di I/O durante l'esecuzione dei programmi degli interrupt di I/O	No	No	No	Impostata nella configurazione del PLC	No
		Abilitazione/disabilitazione interrupt di spegnimento	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	No	Impostata nella configurazione del PLC	No
		Impostazione intervallo degli interrupt programmati	Impostata nella configurazione del PLC (10 ms e 1,0 ms; 0,1 ms solo per le CPU CJ1M)	Impostata nella configurazione del PLC (10 ms e 1,0 ms)	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC (10 ms, 1 ms e 0,5 ms)	No

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Impostazioni iniziali (continua)	Alimentazione	Ritenzione del bit di continuazione al riavvio	No	No	No	Impostata nella configurazione del PLC	No
		Modalità di avvio	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC
		Impostazione delle condizioni di avvio	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
		Registrazione all'avvio	No	No	No	Impostata nella configurazione del PLC	No
		Rilevamento di batteria scarica	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC
		Tempo di rilevamento della caduta di tensione	No	No	No	Impostata nella configurazione del PLC	No
		Ritardo di rilevamento della caduta di tensione	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC (tempo di continuazione del funzionamento dopo il rilevamento della caduta di tensione)	No	No
		Caduta di tensione considerata come errore fatale/non fatale	No	No	No	Impostata nella configurazione del PLC	No
	Cicli	Aggiornamento degli I/O	No	No	Impostato nella configurazione del PLC (solo Moduli di I/O speciale)	Impostata nella configurazione del PLC	No
		Tempo di ciclo costante	Impostato nella configurazione del PLC (da 1 a 32.000 ms)	Impostato nella configurazione del PLC (da 1 a 32.000 ms)	Impostato nella configurazione del PLC (da 1 a 9.999 ms)	Impostato nella configurazione del PLC (da 1 a 32.000 ms)	Impostato nella configurazione del PLC (da 1 a 9.999 ms)
		Tempo di monitoraggio del ciclo	Impostato nella configurazione del PLC (da 10 a 40.000 ms; impostazione iniziale: fissato a 1.000 ms)	Impostato nella configurazione del PLC (da 10 a 40.000 ms; impostazione iniziale: fissato a 1.000 ms)	Impostato nella configurazione del PLC (da 0 a 99). Unità: 1 s, 10 ms, 100 ms (impostazione iniziale: fissato a 120 ms)	Impostato nella configurazione del PLC (da 10 a 40.000 ms; impostazione iniziale: fissato a 1.000 ms)	Impostato nella configurazione del PLC (da 0 a 99). Unità: 1 s, 10 ms, 100 ms (impostazione iniziale: fissato a 120 ms)
		Disabilitazione rilevamento di superamento del tempo di ciclo	No	No	Impostata nella configurazione del PLC	No	Impostata nella configurazione del PLC
		Esecuzione istruzioni e gestione periferiche in modalità asincrona	No	No	No	Impostata nella configurazione del PLC	No
	Comunicazione seriale	Impostazioni di comunicazione della porta RS-232C	Impostazione del DIP switch per il rilevamento automatico o configurazione del PLC	Impostazione del DIP switch per il rilevamento automatico o configurazione del PLC	Impostazione del DIP switch per i valori predefiniti o configurazione del PLC	Impostazione del DIP switch per i valori predefiniti o configurazione del PLC	Impostazione del DIP switch per i valori predefiniti o configurazione del PLC
		Impostazioni di comunicazione della porta periferiche	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Configurazione del PLC	Impostate tramite DIP switch	Impostata nella configurazione del PLC
		Impostazioni di comunicazione della scheda di comunicazione	No	No	Configurazione del PLC	No	Configurazione del PLC
	Modalità di elaborazione della CPU	Modalità di elaborazione parallela	CJ1: No CJ1-H: Sì CJ1M: No	CS1: No CS1-H: Sì	No	No	No
		Modalità di gestione prioritaria delle periferiche	Sì	Sì	No	No	No
	Gestione di altre periferiche	Tempo di gestione	Impostata nella configurazione del PLC (tempo di gestione delle periferiche fisso)	Impostata nella configurazione del PLC (tempo di gestione delle periferiche fisso)	Impostata nella configurazione del PLC (porta RS-232C integrata, scheda di comunicazione, porta periferiche)	No	Impostata nella configurazione del PLC (porta RS-232C integrata, scheda di comunicazione, porta periferiche)
		Misurazione dell'intervallo di gestione del Modulo CPU bus	No	No	No	Impostata nella configurazione del PLC	No
		Interruzione dell'aggiornamento ciclico dei Moduli di I/O speciale	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	Impostata nella configurazione del PLC	No	No
		Applicazione di collegamento al Modulo CPU bus	No	No	No	Impostata nella configurazione del PLC	No

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Impostazioni iniziali (continua)	Console di programmazione	Lingua della Console di programmazione	Impostata tramite DIP switch	CS1: impostata tramite DIP switch CS1-H: impostata dalla Console di programmazione	Impostata tramite DIP switch	No	Impostata tramite DIP switch
	Errori	Area del log degli errori	No (fissa)	No (fissa)	No (fissa: da DM 6001 a DM 6030)	Impostata nella configurazione del PLC	No (fissa: da DM 6569 a DM 6599)
		Esclusione errori FAL definiti dall'utente dal log degli errori	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
	Funzionamento	Attesa della CPU	No	No	No	Impostata nella configurazione del PLC	No
Area ausiliaria	Flag di condizione	Flag ER, CY, <, >, =, attivazione/disattivazione permanente ecc.	Immissione tramite simboli, ad esempio ER	Immissione tramite simboli, ad esempio ER	Sì	Sì	Sì
		Impulsi di clock	Immissione tramite simboli, ad esempio 0,1 s	Immissione tramite simboli, ad esempio 0,1 s	Sì	Sì	Sì
	Gestione	Bit di disabilitazione gestione della CPU	No	No	No	Sì	No
		Codici per dispositivi collegati	No	No	No	Sì	No
		Tempi di ciclo di elaborazione periferiche	No	No	No	Sì	No
		Intervallo di gestione del Modulo CPU bus	No	No	No	Sì	No
		Abilitazione/disabilitazione periferiche collegate alla CPU	No	No	No	Sì	No
		Bit di disabilitazione gestione Host Link/NT Link	No	No	No	Sì	No
		Bit di disabilitazione gestione delle periferiche	No	No	No	Sì	No
		Bit di disabilitazione aggiornamento programmato	No	No	No	Sì	No
		Area di monitoraggio generico della scheda interna	No	Sì	Sì	No	Sì
		Superamento del tempo di ciclo	Sì	Sì	Sì	Sì	Sì
	Task	Flag di primo task	Sì	Sì	No (solo flag di prima scansione)	No (solo flag di prima scansione)	No (solo flag di prima scansione)
	Debug	Flag di modifica in linea disabilitata	Sì	Sì	Sì (AR)	No	No
		Flag di attesa della modifica in linea	Sì	Sì	Sì (AR)	No	No
		Bit di disattivazione uscite	Sì	Sì	Sì	Sì	Sì
		Bit di ritenzione dello stato forzato	Sì	Sì	Sì	Sì	Sì
	Memoria file	Flag delle istruzioni per la memoria file	Sì	Sì	No	Sì	No
		Flag di errore di formattazione della memoria file nell'area EM	Sì (ad eccezione delle CPU CJ1M)	Sì	No	No	No
		Banco di inizio per la formattazione della memoria file nell'area EM	Sì (ad eccezione delle CPU CJ1M)	Sì	No	No	No
	Memoria	Flag di stato del DIP switch	Sì (pin 6)	Sì (pin 6)	Sì (AR, solo pin 6)	No	Sì (AR, pin 6)
		bit di ritenzione IOM	Sì	Sì	Sì	Sì	Sì
	Interrupt	Tempo massimo di elaborazione operazione/subroutine	Sì	Sì	Sì	No	No
		Flag di errore task ad interrupt	Sì	Sì	Sì	No	No

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Area ausiliaria (continua)	Errori	Puntatore/area di memorizzazione del log degli errori	Sì	Sì	No	Sì	No
		Codici di errore	Sì	Sì	Sì	Sì	Sì
	Impostazioni iniziali	Inizializzazione della configurazione del PLC	No	No	Sì	No	Sì
	Comunicazione	Flag di livello operativo per il collegamento al PLC	Sì (bit dell'area ausiliaria per il collegamento al PLC)	Sì (bit dell'area ausiliaria per il collegamento al PLC)	Sì (AR)	No	No
	Alimentazione	Flag di caduta di tensione	No	No	No	Sì	No
		Ora caduta di tensione	No	No	No	Sì	No
		Tempo di esercizio	Sì	Sì	No	Sì	No
		Ora caduta di tensione (spegnimento incluso)	Sì	Sì	No	Sì	Sì
		Numero di cadute di tensione	Sì (numero di cadute di tensione)	Sì (numero di cadute di tensione)	Sì (numero di cadute di tensione)	Sì	Sì (numero di cadute di tensione)
		Tempo di esercizio totale	Sì	Sì	No	No	No
Metodi di assegnazione	Formato		L'assegnazione è basata sul numero di word richieste dai Moduli in ordine di collegamento.	L'assegnazione è basata sul numero di word richieste dai Moduli e gli slot vuoti vengono ignorati.	Assegnazione fissa delle word: a ciascun Modulo viene automaticamente assegnata una word.	L'assegnazione è basata sul numero di word richieste dai Moduli e gli slot vuoti vengono ignorati.	L'assegnazione è basata sul numero di word richieste dai Moduli in ordine di collegamento.
	Assegnazione per i Moduli di I/O ad alta densità del gruppo 2		Nessuno	Come per i Moduli di I/O di base	Area di assegnazione per il gruppo 2 nell'area IR (posizione determinata dal selettore sul pannello frontale)	Nessuno	Nessuno
	Metodo di prenotazione delle word		Modifica della tabella di I/O da CX-Programmer	Modifica della tabella di I/O da CX-Programmer	Creazione di una tabella di I/O a slot vuoto o modifica della tabella di I/O da CX-Programmer	Modulo di I/O fittizio o modifica della tabella di I/O da CX-Programmer	Assegnazione automatica all'avvio
	Assegnazione per i Moduli di I/O speciale	Area CIO	Assegnazione nell'area dei Moduli di I/O speciale in base al numero di Modulo; 10 word per Modulo per un totale di 96 Moduli	Assegnazione nell'area dei Moduli di I/O speciale in base al numero di Modulo; 10 word per Modulo per un totale di 96 Moduli	Assegnazione nell'area dei Moduli di I/O speciale (nell'area IR) in base al numero di Modulo; 10 word per Modulo per un totale di 16 Moduli	Come per i Moduli di I/O di base; 2 o 4 word assegnati nell'area degli I/O (differente per ciascun Modulo)	Come per i Moduli di I/O di base; 1, 2 o 4 word assegnati nell'area degli I/O (differente per ciascun Modulo)
		Area DM	Assegnazione nell'area DM da D20000 a D29599 in base al numero di modulo; 100 word per Modulo per un totale di 96 Moduli	Assegnazione nell'area DM da D20000 a D29599 in base al numero di modulo; 100 word per Modulo per un totale di 96 Moduli	Assegnazione nell'area da DM 1000 a DM 1999 e da DM 2000 a DM 2599; 100 word per Modulo per un totale di 16 Moduli	Nessuno	Nessuno
	Assegnazione per i Moduli CPU bus	Area CIO	Assegnazione nell'area dei Moduli CPU bus in base al numero di Modulo; 25 word per Modulo per un totale di 16 Moduli	Assegnazione nell'area dei Moduli CPU bus in base al numero di Modulo; 25 word per Modulo per un totale di 16 Moduli	Nessuno	Assegnazione nell'area dei Moduli CPU bus in base al numero di Modulo; 25 word per Modulo per un totale di 16 Moduli	Nessuno
		Area DM	Assegnazione nell'area DM da D30000 a D31599 in base al numero di Modulo; 100 word per Modulo per un totale di 16 Moduli	Assegnazione nell'area DM da D30000 a D31599 in base al numero di Modulo; 100 word per Modulo per un totale di 16 Moduli	Nessuno	Assegnazione nell'area DM da D02000 a D03599 in base al numero di Modulo; 100 word per Modulo per un totale di 16 Moduli	Nessuno

Elemento			Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Memoria I/O	Area CIO		Sì	Sì	Sì	Sì	Sì
	Area WR		Sì	Sì	No	No	No
	Area di memorizzazione temporanea		Sì	Sì	Sì	Sì	Sì
	Area ausiliaria		Sì	Sì	Sì	Sì	Sì
	Area SR		No	No	Sì	No	Sì
	Area dei data link		Sì (area dei data link)	Sì (area dei data link)	Sì (area dei data link)	No	Sì
	Area dei Moduli di I/O speciale C200H		Sì	Sì	Sì (area CIO)	No	No
	Area degli I/O integrati		Sì (solo CPU CJ1M con I/O integrati)	No	No	No	No
	Area di collegamento seriale tra PLC		Sì (solo CPU CJ1M)	No	No	No	No
	Area DM		Sì	Sì	Sì	Sì	Sì
	Area di memoria dei dati estesa (EM)		Sì (è possibile specificare gli indirizzi, incluso il numero di banco; non supportata dalla CPU CJ1M)	Sì (è possibile specificare gli indirizzi, incluso il numero di banco)	Sì (è possibile specificare gli indirizzi per -Z, esclusi i banchi)	Sì (non è possibile specificare l'indirizzo né il banco; il banco deve essere modificato. È necessario un Modulo EM)	Sì (banchi esclusi)
	Area temporizzatore/ contatore		Sì	Sì	Sì	Sì	Sì
	Registri indice		Sì	Sì	No	Sì	No
	Registri dati		Sì	Sì	No	Sì	No
	Aree di impostazione/ ripristino forzato	Area CIO	Sì	Sì	Sì	Sì	Nessuno
		Area WR	Sì	Sì	No	No	Sì
		Area di ritenzione	Sì	Sì	Sì	No	No
		Area ausiliaria	No	No	Sì	No	Sì
		Area SR	No	No	No	No	No
		Area dei data link	No	No	Sì	No	No
		Temporizzatori/contatori	Sì (flag)	Sì (flag)	Sì (flag)	Sì (flag)	Sì (flag)
		Area DM	No	No	No	No	No
		Area EM	No	No	No	No	No
Variazioni di istruzioni/indirizzamento indiretto	Differenziazione up (eseguita una volta)		Sì (specificata da @)	Sì (specificata da @)	Sì (specificata da @)	Sì (specificata da ↑)	Sì (specificata da @)
	Differenziazione down (eseguita una volta)		Sì (specificata da %)	Sì (specificata da %)	No (viene utilizzata l'istruzione DIFD)	Sì (specificata da ↓)	No (ottenuta utilizzando DIFD)
	Aggiornamento immediato		Sì (specificato da !)	Sì (specificato da !)	No (viene utilizzata l'istruzione IORF)	Sì (specificato da !)	No (ottenuto utilizzando IORF)
	Indirizzamento indiretto dell'area DM/ EM	Modalità BCD	Sì (da 0000 a 9999) Viene utilizzato l'asterisco.	Sì (da 0000 a 9999) Viene utilizzato l'asterisco.	Sì (da 0 a 9999)	Sì (da 0 a 9999)	Sì (da 0000 a 9999) Viene utilizzato l'asterisco.
		Modalità binaria	Sì (da 00000 a 32767) Viene utilizzato il simbolo @. Da 0000 a 7FFF esadecimale: da 0000 a 31767 Da 8000 a FFFF esadecimale: da 00000 a 32767 nel banco successivo	Sì (da 00000 a 32767) Viene utilizzato il simbolo @. Da 0000 a 7FFF esadecimale: da 0000 a 31767 Da 8000 a FFFF esadecimale: da 00000 a 32767 nel banco successivo	No	Sì, solo per l'indirizzamento indiretto utilizzando gli indirizzi di memoria del PLC	No
Metodi di assegnazione	Impostazione della prima word su un sistema		Sì (per tutte le CPU)	Sì (per tutte le CPU)	No	No	No
	Impostazione della prima word per uno slot		CPU di versioni precedenti alla 2.0: vedere nota. CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: vedere nota. CPU versione 2.0 o successiva: Sì	No	No	No

Elemento		Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Connessioni in linea tramite reti senza creazione di tabelle degli I/O		Con assegnazione degli I/O automatica all'avvio: Sì (per tutte le CPU) Per il funzionamento manuale CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	Sì, ma solo per Controller Link
Utilizzo su più livelli di rete		CPU di versioni precedenti alla 2.0: 3 livelli CPU versione 2.0 o successiva: 8 livelli	CPU di versioni precedenti alla 2.0: 3 livelli CPU versione 2.0 o successiva: 8 livelli	No	Sì, per 3 livelli	No
Connessioni in linea ai PLC tramite PT della serie NS		CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
Protezione da scrittura dai comandi FINS inviati alle CPU tramite reti		CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
Scaricamento di singoli task		CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
Protezione da lettura mediante password	Intero programma utente	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	Task specificati	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	Abilitazione/Disabilitazione creazione di file di programma nella memoria file	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	Protezione da scrittura del programma	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
Trasferimento automatico all'accensione del sistema senza un file dell'area dei parametri (.STD)		CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No

Nota Per le CPU prodotte dall'1 giugno 2002 (numeri di lotto 020601□□□□ o successivi) è possibile impostare fino a otto indirizzi di slot.

Confronto delle istruzioni

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di ingresso sequenza	LOAD/AND/OR	LD/ AND/ OR	Sì	Sì	Sì	Sì	Sì
	AND LOAD/OR LOAD	AND LD/OR LD	Sì	Sì	Sì	Sì	Sì
	NOT	NOT	Sì	Sì	Sì	Sì	No
	CONDITION ON	UP	Sì	Sì	No	Sì (*1)	No
	CONDITION OFF	DOWN	Sì	Sì	No	Sì (*1)	No
	BIT TEST	TST/ TSTN	Sì (posizione del bit specificata in formato binario: da 0000 a 000F esadecimale)	Sì (posizione del bit specificata in formato binario: da 0000 a 000F esadecimale)	Sì (posizione del bit specificata in formato BCD: (*2)	Sì (posizione del bit specificata in formato BCD: (*1)	No
Istruzioni di uscita sequenza	OUTPUT	OUT	Sì	Sì	Sì	Sì	Sì
	TR	TR	Sì	Sì	Sì	Sì	Sì
	KEEP	KEEP	Sì	Sì	Sì	Sì	Sì
	DIFFERENTIATE UP/DOWN	DIFU/ DIFD	Sì (LD↑, AND↑, OR↑) (LD↓, AND↓, OR↓)	Sì (LD↑, AND↑, OR↑) (LD↓, AND↓, OR↓)	Sì (DIFU/DIFD)	Sì (LD↑, AND↑, OR↑)/ (LD↓, AND↓, OR↓)	Sì (DIFU/DIFD)
	SET e RESET	SET/ RSET	Sì	Sì	Sì	Sì	Sì
	MULTIPLE BIT SET/RESET	SETA/ RSTA	Sì (bit di inizio e numero di bit specificati in formato binario)	Sì (bit di inizio e numero di bit specificati in formato binario)	No	(*1) (bit di inizio e numero di bit specificati in formato BCD)	No
	SINGLE BIT SET/ RESET	SET/ RSTB	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
	SINGLE BIT OUTPUT	OUTB	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
Istruzioni di controllo sequenza	END/NO OPERATION	END/ NOP	Sì	Sì	Sì	Sì	Sì
	INTERLOCK/ INTERLOCK CLEAR	IL/ILC	Sì	Sì	Sì	Sì	Sì
	Istruzioni di interblocco multiple	MILH/ MILR/ MILC	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	JUMP/JUMP END	JMP/ JME	Sì (numero di salto specificato in formato BCD: da 0 a 1023)	Sì (numero di salto specificato in formato BCD: da 0 a 1023)	Sì (numero di salto specificato in formato BCD: da 0 a 99)	Sì (numero di salto specificato in formato BCD: da 0 a 999)	Sì (numero di salto specificato in formato BCD: da 0 a 99)
	CONDITIONAL JUMP NOT	CJP/ CJPN	Sì (numero di salto specificato in formato BCD: da 0 a 1023)	Sì (numero di salto specificato in formato BCD: da 0 a 1023)	No	Sì (numero di salto specificato in formato BCD: da 0 a 999) (*1)	No
	MULTIPLE JUMP/ JUMP END	JMP0/ JME0	Sì	Sì	No	No (è tuttavia possibile impostare la configurazione del PLC per l'abilitazione di più salti con numero di salto 0)	No
	FOR-NEXT LOOPS	FOR/ NEXT	Sì	Sì	No	No	No
	BREAK LOOP	BREAK	Sì	Sì	No	No	No

Elemento		Codice mnemonico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di temporizzatore e contatore	TIMER	TIM (BCD)	Sì	Sì	Sì	Sì	Sì
		TIMX (binario)	Sì (*4)	Sì (*4)	No	No	No
	HIGH-SPEED TIMER	TIMH (BCD)	Sì	Sì	Sì	Sì	Sì
		TIMHX (binario)	Sì (*4)	Sì (*4)	No	No	No
	ONE-MS TIMER	TMHH (BCD)	Sì	Sì	No	No	No
		TMHHX (binario)	Sì (*4)	Sì (*4)	No	No	No
	ACCUMULATIVE TIMER	TTIM (BCD)	Sì	Sì	Sì	Sì	Sì
		TTIMX (binario)	Sì (*4)	Sì (*4)	No	No	No
	LONG TIMER	TIML (BCD)	Sì	Sì	No	Sì	No
		TIMLX (binario)	Sì (*4)	Sì (*4)	No	No	No
	MULTI-OUTPUT TIMER	MTIM (BCD)	Sì	Sì	No	Sì	No
		MTIMX (binario)	Sì (*4)	Sì (*4)	No	No	No
	COUNTER	CNT (BCD)	Sì	Sì	Sì	Sì	Sì
		CNTX (binario)	Sì (*4)	Sì (*4)	No	No	No
	REVERSIBLE COUNTER	CNTR (BCD)	Sì	Sì	Sì	Sì	Sì
		CNTRX (binario)	Sì (*4)	Sì (*4)	No	No	No
	RESET TIMER/ COUNTER	CNR (BCD)	Sì (viene reimpostato solo il temporizzatore o il contatore)	Sì (viene reimpostato solo il temporizzatore o il contatore)	No	Sì (viene azzerato anche l'intervallo specificato nell'area CIO)	No
		CNRX (binario)	Sì (*4)	Sì (*4)	No	No	No

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di confronto	Confronto con simboli	=, < ecc.	Sì (sono tutti supportati per LD, OR e AND)	Sì (sono tutti supportati per LD, OR e AND)	Sì (*2) (supportati solo per AND)	Sì (*1) (supportati solo per AND)	No
	Istruzioni di confronto dei dati	=DT, <DT, ecc.	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	COMPARE/ DOUBLE COMPARE	CMP/ CMPL	Sì	Sì	Sì	Sì (*3)	Sì
	SIGNED BINARY COMPARE/ DOUBLE SIGNED BINARY COMPARE	CPS/ CPSL	Sì	Sì	Sì	Sì (*1)	Sì
	BLOCK COMPARE	BCMP	Sì	Sì	Sì	Sì	Sì
	EXTENDED BLOCK COMPARE	BCMP2	Sì (solo CPU CJ1M e CJ1-H versione 2.0)	Sì (solo CPU CJ1-H/CS1D versione 2.0)	No	No	No
	TABLE COMPARE	TCMP	Sì	Sì	Sì	Sì	Sì
	MULTIPLE COMPARE	MCMP	Sì	Sì	Sì	Sì	Sì
	EQUALS	EQU	No	No	No	Sì	No
	AREA RANGE COMPARE	ZCP/ ZCPL	CJ1: No (ottenuto utilizzando istruzioni di confronto) CJ1-H: Sì CJ1M: Sì	CS1: No (ottenuto utilizzando istruzioni di confronto) CS1-H: Sì	Sì	No	No (ottenuto utilizzando istruzioni di confronto)

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di sposta-mento dei dati	MOVE	MOV	Sì	Sì	Sì	Sì	Sì
	DOUBLE MOVE	MOVL	Sì	Sì	No	Sì	No
	MOVE NOT	MVN	Sì	Sì	Sì	Sì	Sì
	DOUBLE MOVE	MVNL	Sì	Sì	No	Sì	No
	DATA EXCHANGE	XCHG	Sì	Sì	Sì	Sì	Sì
	DOUBLE DATA EXCHANGE	XCGL	Sì	Sì	No	Sì	No
	MOVE QUICK	MOVQ	No	No	No	Sì	No
	BLOCK TRANSFER	XFER	Sì (numero di word da trasferire specificato in formato binario: da 0 a 65535)	Sì (numero di word da trasferire specificato in formato binario: da 0 a 65535)	Sì (numero di word da trasferire specificato in formato BCD: da 0 a 6144)	Sì (numero di word da trasferire specificato in formato BCD: da 0 a 9999)	Sì (numero di word da trasferire specificato in formato BCD: da 0 a 9999)
	BLOCK SET	BSET	Sì	Sì	Sì	Sì	Sì
	MOVE BIT	MOVB	Sì (posizioni del bit di origine e del bit di destinazione specificate in formato binario)	Sì (posizioni del bit di origine e del bit di destinazione specificate in formato binario)	Sì (posizioni del bit di origine e del bit di destinazione specificate in formato BCD)	Sì (posizioni del bit di origine e del bit di destinazione specificate in formato BCD)	Sì (posizioni del bit di origine e del bit di destinazione specificate in formato BCD)
	MULTIPLE BIT TRANSFER	XFRB	Sì	Sì	Sì	Sì (*1)	Sì
	MOVE DIGIT	MOVD	Sì	Sì	Sì	Sì	Sì
	SINGLE WORD DISTRIBUTE	DIST	Sì (la funzione di operazione sullo stack è possibile con un'altra istruzione. Valore di offset specificato in formato binario: da 0 a 65535)	Sì (la funzione di operazione sullo stack è possibile con un'altra istruzione. Valore di offset specificato in formato binario: da 0 a 65535)	Sì (la funzione di operazione sullo stack è possibile. Valore di offset specificato in formato BCD: da 0 a 8999)	Sì (la funzione di operazione sullo stack è possibile con un'altra istruzione. Valore di offset specificato in formato BCD: da 0 a 9999)	Sì (la funzione di operazione sullo stack è possibile. Valore di offset specificato in formato BCD: da 0 a 8999)
	DATA COLLECT	COLL	Sì (la funzione di operazione sullo stack è possibile con un'altra istruzione. Valore di offset specificato in formato binario: da 0 a 65535)	Sì (la funzione di operazione sullo stack è possibile con un'altra istruzione. Valore di offset specificato in formato binario: da 0 a 65535)	Sì (la funzione di operazione sullo stack è possibile. Valore di offset specificato in formato BCD: da 0 a 7999)	Sì (la funzione di operazione sullo stack è possibile con un'altra istruzione. Valore di offset specificato in formato BCD: da 0 a 9999)	Sì (la funzione di operazione sullo stack è possibile. Valore di offset specificato in formato BCD: da 0 a 7999)
	EM BLOCK TRANSFER BETWEEN BANKS	BXFR	No (funzionalmente possibile per un massimo di 65.535 word indirizzando direttamente l'area EM tramite XFER)	No (funzionalmente possibile per un massimo di 65.535 word indirizzando direttamente l'area EM tramite XFER)	No	Sì (*1)	No
	EM BLOCK TRANSFER	XFR2	No	No	Sì	No	No
	EM BANK TRANSFER	BXF2	No	No	Sì	No	No
	MOVE TO REGISTER	MOVR	Sì (non viene specificato alcun indirizzo per l'area DM/EM indiretta)	Sì (non viene specificato alcun indirizzo per l'area DM/EM indiretta)	No	Sì (l'indirizzo viene specificato per l'area EM/DM indiretta)	No
	MOVE TIMER/COUNTER PV TO REGISTER	MOVRW	Sì	Sì	No	No (possibile per i flag di completamento solo utilizzando MOVR)	No

Elemento		Codice mnemonico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di scorrimento dei dati	SHIFT REGISTER	SFT	Sì	Sì	Sì	Sì	Sì
	REVERSIBLE SHIFT REGISTER	SFTR	Sì	Sì	Sì	Sì	Sì
	ASYNCHRONOUS SHIFT REGISTER	ASFT	Sì	Sì	Sì	Sì	Sì
	WORD SHIFT	WSFT	Sì (come per CV: 3 operandi)	Sì (come per CV: 3 operandi)	Sì	Sì	Sì
	ARITHMETIC SHIFT LEFT/ ARITHMETIC SHIFT RIGHT	ASL/ ASR	Sì	Sì	Sì	Sì	Sì
	ROTATE LEFT/ ROTATE RIGHT	ROL/ ROR	Sì	Sì	Sì	Sì	Sì
	ONE DIGIT SHIFT LEFT/ONE DIGIT SHIFT RIGHT	SLD/ SRD	Sì	Sì	Sì	Sì	Sì
	SHIFT N-BIT DATA LEFT/SHIFT N-BIT DATA RIGHT	NSFR/ NSFL	Sì (dati di scorrimento e bit di inizio specificati in formato binario)	Sì (dati di scorrimento e bit di inizio specificati in formato binario)	No	Sì (dati di scorrimento e bit di inizio specificati in formato BCD) (*1)	No
	SHIFT N-BITS LEFT/SHIFT N-BITS RIGHT/ DOUBLE SHIFT N-BITS LEFT/ DOUBLE SHIFT N-BITS RIGHT	NASL/ NASR, NSLL/ NSRL	Sì (numero di bit da far scorrere specificato in formato binario)	Sì (numero di bit da far scorrere specificato in formato binario)	No	Sì (numero di bit da far scorrere specificato in formato BCD) (*1)	No
	DOUBLE SHIFT LEFT/DOUBLE SHIFT RIGHT	ASLL/ ASRL	Sì	Sì	No	Sì	No
	DOUBLE ROTATE LEFT/DOUBLE ROTATE RIGHT	ROLL/ RORL	Sì	Sì	No	Sì	No
	ROTATE LEFT WITHOUT CARRY/ROTATE RIGHT WITHOUT CARRY/DOUBLE ROTATE LEFT WITHOUT CARRY/DOUBLE ROTATE RIGHT WITHOUT CARRY	RLNC/ RRNC, RLNL/ RRNL	Sì	Sì	No	Sì (*1)	No
Istruzioni di incremento e decremento	INCREMENT BCD/ DECREMENT BCD	++B/– –B (INC/ DEC)	Sì (++B/– –B)	Sì (++B/– –B)	Sì (INC/DEC)	Sì (INC/DEC)	Sì (INC/DEC)
	DOUBLE INCREMENT BCD/ DOUBLE DECREMENT BCD	++BL/– –BL (INCL/ DECL)	Sì (++BL/– –BL)	Sì (++BL/– –BL)	No	Sì (INCL/DECL)	No
	INCREMENT BINARY/ DECREMENT BINARY	++/– – (INCB/ DECB)	Sì (CY viene attivato per il riporto o il prestito) (++/– –)	Sì (CY viene attivato per il riporto o il prestito) (++/– –)	No	Sì	No
	DOUBLE INCREMENT BINARY/DOUBLE DECREMENT BINARY	++L/– –L (INBL/ DCBL)	Sì (CY viene attivato per il riporto o il prestito) (++L/– –L)	Sì (CY viene attivato per il riporto o il prestito) (++L/– –L)	No	Sì	No
Istruzioni matematiche			Sì	Sì	Sì	Sì	Sì

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di conver-sione	BCD-TO-BINARY/ DOUBLE BCD-TO-DOUBLE BINARY	BIN/ BINL	Sì	Sì	Sì	Sì	Sì
	BINARY-TO-BCD/ DOUBLE BINARY-TO-DOUBLE BCD	BCD/ BCDL	Sì	Sì	Sì	Sì	Sì
	2'S COMPLEMENT/ DOUBLE 2'S COMPLEMENT	NEG/ NEGL	Sì (come per CV ma UP non viene attivato per 8000 esadecimal e nell'origine)	Sì (come per CV ma UP non viene attivato per 8000 esadecimal e nell'origine)	Sì	Sì	Sì
	16-BIT TO 32-BIT SIGNED BINARY	SIGN	Sì	Sì	No	Sì	No
	DATA DECODER	MLPX	Sì	Sì	Sì	Sì	Sì
	DATA ENCODER	DMPX	Sì (come per CVM1-V2: è pos-sibile specificare il bit all'estrema destra per ON)	Sì (come per CVM1-V2: è pos-sibile specificare il bit all'estrema destra per ON)	Sì (bit all'estrema sinistra solo per ON)	Sì (CVM1-V2: è possibile specifi-care il bit all'estrema destra per ON)	Sì (bit all'estrema sinistra solo per ON)
	ASCII CONVERT	ASC	Sì	Sì	Sì	Sì	Sì
	ASCII TO HEX	HEX	Sì	Sì	Sì	Sì (*1)	Sì
	COLUMN TO LINE/LINE TO COLUMN	LINE/ COLM	Sì (posizione del bit specificata in formato binario)	Sì (posizione del bit specificata in formato binario)	Sì (posizione del bit specificata in formato BCD)	Sì (posizione del bit specificata in formato BCD)	Sì (posizione del bit specificata in formato BCD)
	SIGNED BCD-TO-BINARY/DOUBLE SIGNED BCD-TO-BINARY	BINS/ BISL	Sì	Sì	No	Sì (*1)	No
	SIGNED BINARY-TO-BCD/DOUBLE SIGNED BINARY-TO-BCD	BCDS/ BDSL	Sì	Sì	No	Sì (*1)	No
	GRAY CODE CONVERSION	GRY	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì (supportato anche per i numeri di lotto 030201 o suc-cessivi)	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì (supportato anche per i numeri di lotto 030201 o suc-cessivi)	No	No	No
Istruzioni logiche	LOGICAL AND/ LOGICAL OR/ EXCLUSIVE OR/ EXCLUSIVE NOR	ANDW, ORW, XORW, XNRW	Sì	Sì	Sì	Sì	Sì
	DOUBLE LOGI-CAL AND/DOU-BLE LOGICAL OR/ DOUBLE EXCLU-SIVE OR/DOUBLE EXCLUSIVE NOR	ANDL, ORWL, XORL, XNRL	Sì	Sì	No	Sì	No
	COMPLEMENT/ DOUBLE COMPLEMENT	COM/ COML	Sì	Sì	Sì (solo COM)	Sì	Sì (solo COM)

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni matemati-che speciali	BCD SQUARE ROOT	ROOT	Sì	Sì	Sì	Sì	Sì
	BINARY ROOT	ROTB	Sì	Sì	No	Sì (*1)	No
	ARITHMETIC PROCESS	APR	Sì	Sì	Sì	Sì	Sì
	FLOATING POINT DIVIDE	FDIV	Sì	Sì	Sì	Sì	No
	BIT COUNTER	BCNT	Sì (numero di word da conteg-giare e risultati del conteggio specificati in for-mato binario: da 0 a FFFF esa-decimale)	Sì (numero di word da conteg-giare e risultati del conteggio specificati in for-mato binario: da 0 a FFFF esa-decmale)	Sì (numero di word da conteg-giare e risultati del conteggio specificati in for-mato BCD: da 1 a 6656)	Sì (numero di word da conteg-giare e risultati del conteggio specifi-cati in formato BCD: da 0 a 9999, ma si veri-fica errore per 0)	Sì (numero di word da conteg-giare e risultati del conteggio specificati in for-mato BCD: da 1 a 6656)
Istruzioni matemati-che a vir-gola mobile	FLOATING TO 16-BIT/32-BIT BIN, 16-BIT/32-BIT BIN TO FLOATING	FIX/ FIXL, FLT/ FLTL	Sì	Sì	No	Sì (*1)	Sì
	FLOATING-POINT ADD/FLOATING-POINT SUBTRACT/ FLOATING-POINT MULTIPLY/ FLOATING-POINT DIVIDE	+F, –F, *F, /F	Sì	Sì	No	Sì (*1)	Sì
	DEGREES TO RADIANS/ RADIANS TO DEGREES	RAD, DEG	Sì	Sì	No	Sì (*1)	Sì
	SINE/COSINE/ TANGENT/ARC SINE/ARC TANGENT	SIN, COS, TAN, ASIN, ACOS, ATAN	Sì	Sì	No	Sì (*1)	Sì
	SQUARE ROOT	SQRT	Sì	Sì	No	Sì (*1)	Sì
	EXPONENT	EXP	Sì	Sì	No	Sì (*1)	Sì
	LOGARITHM	LOG	Sì	Sì	No	Sì (*1)	Sì
	EXPONENTIAL POWER	PWR	Sì	Sì	No	No	No
	Istruzioni di confronto con decimali a virgola mobile	Esempi: =F, <>F	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
	Istruzioni di conversione da decimale a virgola mobile in stringa di testo	FSTR, FVAL	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
Istruzioni di calcolo e conversion e a virgola mobile in doppia precisione	Uguali alle istruzioni di calcolo e conversione a virgola mobile in singola precisione sopra riportate.	Esempi o FIXD	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di elabora-zione dei dati delle tabelle	SET STACK	SSET	Sì (quattro word di dati di con-trollo dello stack. Numero di word specificato in for-mato binario: da 5 a 65535)	Sì (quattro word di dati di con-trollo dello stack. Numero di word specificato in for-mato binario: da 5 a 65535)	No	Sì (quattro word di dati di con-trollo dello stack. Numero di word specificato in for-mato BCD: da 3 a 9999)	No
	PUSH ONTO STACK	PUSH	Sì	Sì	No	Sì	No
	FIRST IN FIRST OUT	FIFO	Sì	Sì	No	Sì	No
	LAST IN FIRST OUT	LIFO	Sì	Sì	No	Sì	No
	FIND MAXIMUM/ FIND MINIMUM	MAX, MIN	Sì (due word nel campo dei dati di controllo. Lun-ghezza della tabella specifi-cata in formato binario: da 1 a FFFF)	Sì (due word nel campo dei dati di controllo. Lun-ghezza della tabella specifi-cata in formato binario: da 1 a FFFF)	Sì (una word nel campo dei dati di controllo. Lun-ghezza della tabella specifi-cata in formato BCD: da 1 a 999)	Sì (una word nel campo dei dati di controllo. Lun-ghezza della tabella specifi-cata in formato BCD: da 1 a 999)	Sì (una word nel campo dei dati di controllo. Lun-ghezza della tabella specifi-cata in formato BCD: da 1 a 999)
	DATA SEARCH	SRCH	Sì (lunghezza della tabella spe-cificata in formato binario: da 1 a FFFF. Indirizzo di memoria del PLC inviato a IR0. È possibile inviare il numero di corri-spondenze a DR0)	Sì (lunghezza della tabella spe-cificata in formato binario: da 1 a FFFF. Indirizzo di memoria del PLC inviato a IR0. È possibile inviare il numero di corri-spondenze a DR0)	Sì (lunghezza della tabella spe-cificata in formato BCD: da 1 a 6556. Indirizzo di memoria del PLC inviato a C+1. Non è possibile inviare il numero di corrispon-denze a DR0)	Sì (lunghezza della tabella spe-cificata in formato BCD: da 1 a 9999. Indirizzo di memoria del PLC inviato a IR0. Non è possibile inviare il numero di corrispon-denze a DR0)	Sì (lunghezza della tabella spe-cificata in formato BCD: da 1 a 6556. Indirizzo di memoria del PLC inviato a C+1. Non è possibile inviare il numero di corrispon-denze a DR0)
	FRAME CHECKSUM	FCS	Sì	Sì	Sì	No	Sì
	SUM	SUM	Sì (come per C200HX/HG/ HE: somma possibile per byte e word)	Sì (come per C200HX/HG/ HE: somma possibile per byte e word)	Sì (somma possibile per byte e word)	Sì (somma possibile solo per word)	Sì (somma possibile per byte e word)
	SWAP BYTES	SWAP	Sì (può essere utilizzata per lo scambio di dati e altre applicazioni)	Sì (può essere utilizzata per lo scambio di dati e altre applicazioni)	No	No	No
	DIMENSION RECORD TABLE	DIM	Sì	Sì	No	No	No
	SET RECORD LOCATION	SETR	Sì	Sì	No	No	No
	GET RECORD LOCATION	GETR	Sì	Sì	No	No	No

Elemento		Codice mnemonico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di controllo dei dati	SCALING	SCL	Sì	Sì	Sì	No	Sì
	SCALING 2	SCL2	Sì	Sì	No	No	Sì
	SCALING 3	SCL3	Sì	Sì	No	No	Sì
	PID CONTROL	PID	Sì (l'uscita può essere commutata tra 0% e 50% quando PV = SV. PID e periodo di campionamento specificati in formato binario)	Sì (l'uscita può essere commutata tra 0% e 50% quando PV = SV. PID e periodo di campionamento specificati in formato binario)	Sì (PID e periodo di campionamento specificati in formato BCD)	Sì (PID e periodo di campionamento specificati in formato BCD) (*1)	Sì (PID e periodo di campionamento specificati in formato BCD)
	PID CONTROL WITH AUTOTUNIG	PIDAT	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
	LIMIT CONTROL	LMT	Sì	Sì	No	Sì (*1)	No
	DEAD BAND CONTROL	BAND	Sì	Sì	No	Sì (*1)	No
	DEAD ZONE CONTROL	ZONE	Sì	Sì	No	Sì (*1)	No
	TIME-PROPORTIONAL OUTPUT	TPO	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	AVERAGE	AVG	Sì (numero di scansioni specificato in formato binario)	Sì (numero di scansioni specificato in formato binario)	Sì (numero di scansioni specificato in formato BCD)	No	Sì (numero di scansioni specificato in formato BCD)
Istruzioni di subroutine	SUBROUTINE CALL/SUBROUTINE ENTRY/ SUBROUTINE RETURN	SBS, SBN, RET	Sì (numero di subroutine specificato in formato BCD: da 0 a 1023)	Sì (numero di subroutine specificato in formato BCD: da 0 a 1023)	Sì (numero di subroutine specificato in formato BCD: da 0 a 255)	Sì (numero di subroutine specificato in formato BCD: da 0 a 999)	Sì (numero di subroutine specificato in formato BCD: da 0 a 255)
	MACRO	MCRO	Sì (numero di subroutine specificato in formato BCD: da 0 a 1023)	Sì (numero di subroutine specificato in formato BCD: da 0 a 1023)	Sì (numero di subroutine specificato in formato BCD: da 0 a 255)	Sì (numero di subroutine specificato in formato BCD: da 0 a 999) (*1)	Sì (numero di subroutine specificato in formato BCD: da 0 a 255)
	Istruzioni di subroutine globali	GSBS, GSBN, RET	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
Istruzioni di controllo degli interrupt	SET INTERRUPT MASK	MSKS	Sì	Sì	No (elaborazione di tutti gli interrupt eseguita tramite INT)	Sì	No (elaborazione di tutti gli interrupt eseguita tramite INT)
	CLEAR INTERRUPT	CLI	Sì	Sì	No	Sì	No
	READ INTERRUPT MASK	MSKR	Sì	Sì	No	Sì	No
	DISABLE INTERRUPTS	DI	Sì	Sì	No	No	No
	ENABLE INTERRUPTS	EI	Sì	Sì	No	No	No
	ENABLE TIMER	STIM	No	No	No	No	Sì

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni per contatore veloce e uscita a impulsi	MODE CONTROL	INI	Sì (*5)	No	No	No	Sì
	PRESENT VALUE READ	PRV	Sì (*5)	No	No	No	Sì
	COUNTER FREQUENCY CONVERT	PRV2	CPU CJ1M versione 2.0 o successiva: Sì (*5) CJ1-H (tutte le CPU): No	No	No	No	No
	SET COMPARISON TABLE	CTBL	Sì (*5)	No	No	No	Sì
	SET PULSES	PULS	Sì (*5)	No	No	No	Sì
	SET FREQUENCY	SPED	Sì (*5)	No	No	No	Sì
	ACCELERATION CONTROL	ACC	Sì (*5)	No	No	No	Sì
	POSITION CONTROL	PLS2	Sì (*5)	No	No	No	Sì
	ORIGIN SEARCH	ORG	Sì (*5)	No	No	No	No
	PWM OUTPUT	PWM	Sì (*5)	No	No	No	Sì
Istruzioni di step	STEP DEFINE e STEP START	STEP/ SNXT	Sì	Sì	Sì	Sì	Sì

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni per Moduli di I/O di base	I/O REFRESH (aggiornamento degli I/O)	IORF	Sì	Sì (utilizzata per i Moduli di I/O ad alta densità C200H gruppo 2 e i Moduli di I/O speciale. Include la funzionalità GROUP-2 HIGH-DENSITY I/O REFRESH (MPRF))	Sì (utilizzata per i Moduli di I/O ad alta densità C200H gruppo 2 e i Moduli di I/O speciale)	Sì	Sì
	7-SEGMENT DECODER	SDEC	Sì	Sì	Sì	Sì	Sì
	DIGITAL SWITCH INPUT	DSW	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	Sì	No	Sì
	TEN KEY INPUT	TKY	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	Sì	No	Sì
	HEXADECIMAL KEY INPUT	HKY	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	Sì	No	Sì
	MATRIX INPUT	MTR	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	Sì	No	No
	7-SEGMENT DISPLAY OUTPUT	7SEG	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	Sì	No	Sì
	GROUP-2 HIGH-DENSITY I/O REFRESH	MPRF	No	No	Sì	No	No
	TEN KEY INPUT	TKY	No	No	Sì	No	Sì
	HEXADECIMAL KEY INPUT	HKY	No	No	Sì	No	Sì
	DIGITAL SWITCH INPUT	DSW	No	No	Sì	No	Sì
	MATRIX INPUT	MTR	No	No	Sì	No	No
	7-SEGMENT DISPLAY OUTPUT	7SEG	No	No	Sì	No	Sì
Istruzioni per Moduli di I/O speciale	SPECIAL I/O UNIT READ e SPECIAL I/O UNIT WRITE (I/O READ e I/O WRITE)	IORD/ IOWR (READ/ WRIT)	IORD/IOWR (fino a 96 Moduli. Non viene più utilizzata per l'invio di comandi FINS)	IORD/IOWR (fino a 96 Moduli. Non viene più utilizzata per l'invio di comandi FINS)	IORD/IOWR	READ/WRIT	No
	I/O READ 2 e I/O WRITE 2	RD2/ WR2	No	No	No	Sì (*1)	No

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di elaborazione delle stringhe di testo	MOV STRING	MOV$	Sì	Sì	No	No	No
	CONCATENATE STRING	+$	Sì	Sì	No	No	No
	GET STRING LEFT	LEFT$	Sì	Sì	No	No	No
	GET STRING RIGHT	RGHT$	Sì	Sì	No	No	No
	GET STRING MIDDLE	MID$	Sì	Sì	No	No	No
	FIND IN STRING	FIND$	Sì	Sì	No	No	No
	STRING LENGTH	LEN$	Sì	Sì	No	No	No
	REPLACE IN STRING	RPLC$	Sì	Sì	No	No	No
	DELETE STRING	DEL$	Sì	Sì	No	No	No
	EXCHANGE STRING	XCHG$	Sì	Sì	No	No	No
	CLEAR STRING	CLR$	Sì	Sì	No	No	No
	INSERT INTO STRING	INS$	Sì	Sì	No	No	No

Elemento		Codice mnemonico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni per la comunicazione seriale	RECEIVE	RXD	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata solo per la porta RS-232C della CPU. Non può essere utilizzata per il Modulo di comunicazione seriale o la porta periferiche della CPU.	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata solo per la porta RS-232C della CPU. Non può essere utilizzata per la scheda interna, il Modulo di comunicazione seriale o la porta periferiche della CPU.	Sì (numero di byte memorizzati specificato in formato BCD). Utilizzata solo per la porta periferiche, la porta RS-232C o la scheda di comunicazione della CPU.	No	Sì (numero di byte memorizzati specificato in formato BCD). Utilizzata solo per la porta periferiche, la porta RS-232C o la scheda di comunicazione della CPU.
	RECEIVE VIA SERIAL COMMUNICATIONS UNIT	RXDU	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata per moduli di comunicazione seriale versione 1.2 o successiva.	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata per moduli di comunicazione seriale versione 1.2 o successiva.	No	No	No
	TRANSMIT	TXD	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata solo per la porta RS-232C della CPU. Non può essere utilizzata per il Modulo di comunicazione seriale o la porta periferiche della CPU. Le comunicazioni non richieste non sono possibili utilizzando il comando Host Link EX.	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata solo per la porta RS-232C della CPU. Non può essere utilizzata per la scheda interna, il Modulo di comunicazione seriale o la porta periferiche della CPU. Le comunicazioni non richieste non sono possibili utilizzando il comando Host Link EX.	Sì (numero di byte memorizzati specificato in formato BCD). Utilizzata solo per la porta periferiche, la porta RS-232C o la scheda di comunicazione della CPU. Le comunicazioni non richieste sono possibili utilizzando il comando Host Link EX.	No	Sì (numero di byte memorizzati specificato in formato BCD). Utilizzata solo per la porta periferiche, la porta RS-232C o la scheda di comunicazione della CPU. Le comunicazioni non richieste sono possibili utilizzando il comando Host Link EX.
	TRANSMIT VIA SERIAL COMMUNICATIONS UNIT	TXDU	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata per moduli di comunicazione seriale versione 1.2 o successiva.	Sì (numero di byte memorizzati specificato in formato binario). Utilizzata per moduli di comunicazione seriale versione 1.2 o successiva.	No	No	No
	CHANGE SERIAL PORT SETUP	STUP	Sì (10 word impostate). Può essere utilizzata per il Modulo di comunicazione seriale.	Sì (10 word impostate). Può essere utilizzata per il Modulo di comunicazione seriale.	Sì (5 word impostate)	No	Sì (5 word impostate)
	PROTOCOL MACRO	PMCR	Sì (numero di sequenza specificato in formato binario; quattro operandi). È possibile specificare l'indirizzo del modulo di destinazione e il numero di porta seriale.	Sì (numero di sequenza specificato in formato binario; quattro operandi). È possibile specificare l'indirizzo del modulo di destinazione e il numero di porta seriale.	Sì (numero di sequenze specificato in formato BCD; tre operandi)	No	Sì (numero di sequenze specificato in formato BCD; tre operandi)
	PCMCIA CARD MACRO	CMCR	No	No	Sì	No	No

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni di rete	NETWORK SEND/ NETWORK RECEIVE	SEND/ RECV	Sì. Può essere utilizzata per il computer host tramite connessioni Host Link. Non può essere utilizzata per i Moduli di comunicazione seriale o la porta RS-232C della CPU.	Sì. Può essere utilizzata per il computer host tramite connessioni Host Link. Non può essere utilizzata per i Moduli di comunicazione seriale, la porta RS-232C della CPU o la scheda interna.	Sì. Non può essere utilizzata per il computer host tramite connessioni Host Link.	Sì. Può essere utilizzata per il computer host tramite connessioni Host Link.	Sì. Non può essere utilizzata per il computer host tramite connessioni Host Link.
	DELIVER COMMAND	CMND	Sì. Viene utilizzata per il computer host tramite connessioni Host Link. Non può essere utilizzata per i Moduli di comunicazione seriale o la porta RS-232C della CPU.	Sì. Viene utilizzata per il computer host tramite connessioni Host Link. Non può essere utilizzata per i Moduli di comunicazione seriale, la porta RS-232C della CPU o la scheda interna.	No	Sì. Può essere utilizzata per il computer host tramite connessioni Host Link.	Sì. Non può essere utilizzata per il computer host tramite connessioni Host Link.
	EXPLICIT MESSAGE SEND	EXPLT	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	EXPLICIT GET ATTRIBUTE	EGATR	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	EXPLICIT SET ATTRIBUTE	ESATR	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	EXPLICIT WORD READ	ECHRD	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
	EXPLICIT WORD WRITE	ECHWR	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	CPU di versioni precedenti alla 2.0: No CPU versione 2.0 o successiva: Sì	No	No	No
Istruzioni relative alla memoria file	READ DATA FILE/ WRITE DATA FILE	FREAD/ FWRIT	Sì	Sì	No	Sì (FILR/FILW)	No
	READ PROGRAM FILE	FILP	No	No	No	Sì	No
	CHANGE STEP PROGRAM	FLSP	No	No	No	Sì	No
Istruzioni di visualizzazione	DISPLAY MESSAGE	MSG	Sì (messaggi con terminazione NUL)	Sì (messaggi con terminazione NUL)	Sì (messaggi con terminazione CR)	Sì (messaggi con terminazione CR)	Sì (messaggi con terminazione CR)
	DISPLAY LONG MESSAGE	LMSG	No	No	Sì (messaggi con terminazione CR)	No	No
	I/O DISPLAY	IODP	No	No	No	Sì	No
	TERMINAL MODE	TERM	No	No	Sì	No	No

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Istruzioni per l'orologio	CALENDAR ADD	CADD	Sì	Sì	No	Sì	No
	CALENDAR SUBTRACT	CSUB	Sì	Sì	No	Sì	No
	HOURS TO SECONDS	SEC	Sì	Sì	Sì	Sì	Sì
	SECONDS TO HOURS	HMS	Sì	Sì	Sì	Sì	Sì
	CLOCK ADJUSTMENT	DATE	Sì	Sì	No	Sì (*1)	No
Istruzioni di debug	TRACE MEMORY SAMPLING	TRSM	Sì	Sì	Sì	Sì	Sì
	MARK TRACE	MARK	No	No	No	Sì (numero di contrassegno specificato in formato BCD)	No
Istruzioni per la diagnostica di funzionamento incorretto	FAILURE ALARM/ SEVERE FAILURE ALARM	FAL/ FALS	Sì (messaggi con terminazione NUL, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato binario)	Sì (messaggi con terminazione NUL, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato binario)	Sì (messaggi con terminazione CR, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato BCD)	Sì (messaggi con terminazione CR, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato BCD)	Sì (messaggi con terminazione CR, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato BCD)
	FAILURE POINT DETECTION	FPD	Sì (messaggi con terminazione NUL, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato binario)	Sì (messaggi con terminazione NUL, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato binario)	Sì (messaggi con terminazione CR, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato BCD)	Sì (messaggi con terminazione CR, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato BCD) (*1)	Sì (messaggi con terminazione CR, stringhe di testo memorizzate nel seguente ordine: dal byte all'estrema sinistra al byte all'estrema destra, quindi dalla word all'estrema destra alla word all'estrema sinistra; numero di FAL specificato in formato BCD)

Elemento		Codice mne-monico	Serie CJ	Serie CS	C200HX/HG/HE	CVM1/serie CV	CQM1H
Altre istruzioni	SET CARRY/ CLEAR CARRY	STC/ CLC	Sì	Sì	Sì	Sì	Sì
	LOAD FLAGS/ SAVE FLAGS	CCL, CCS	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	Sì	No
	EXTEND MAXIMUM CYCLE TIME	WDT	Sì	Sì	Sì	Sì (*1)	Sì
	CYCLE TIME	SCAN	No	No	Sì	No	No
	LOAD REGISTER/ SAVE REGISTER	REGL, REGS	No	No	No	Sì	No
	SELECT EM BANK	EMBC	Sì	Sì	Sì	Sì	No
	EXPANSION DM READ	XDMR	No	No	Sì	No	No
	INDIRECT EM ADDRESSING	IEMS	No	No	Sì	No	No
	ENABLE ACCESS/ DISABLE ACCESS	IOSP, IORS	No	CS1: No CS1-H: Sì	No	Sì	No
	Istruzioni di conversione degli indirizzi della serie CV in indirizzi della serie CS e viceversa	FRMCV, TOCV	CJ1: No CJ1-H: Sì CJ1M: Sì	CS1: No CS1-H: Sì	No	No	No
Istruzioni di programmazione a blocchi		BPRG/ BEND, IF/ ELSE/ IEND, WAIT, EXIT, LOOP/ LEND, BPPS/ BPRS, TIMW, CNTW, TMHW	Sì	Sì	No	Sì (*1)	No
Istruzioni di controllo dei task	TASK ON/TASK OFF	TKON/ TKOF	Sì	Sì	No	No	No

Nota *1: supportata solo da CVM1 (V2).
*2: supportata solo dai modelli CPU□□-Z.
*3: continuazione nell'ambito dell'esecuzione dello stesso programma supportata da CV1M versione 2,
*4: a eccezione delle CPU CS1 e CJ1.
*5: solo CPU CJ1M con I/O integrati. Alcuni operandi differiscono da quelli utilizzati da CQM1H.

Appendice B
Differenze rispetto ai sistemi Host Link precedenti

Esistono alcune differenze tra i sistemi Host Link creati utilizzando i Moduli di comunicazione seriale della serie CS/CJ o Schede di comunicazione seriale (solo serie CS) rispetto ai sistemi Host Link realizzati mediante Moduli Host Link e CPU di altre serie di PLC. In questa appendice viene fornita una descrizione di tali differenze.

Porte RS-232C

Quando si passa da un sistema Host Link esistente a un altro che utilizza una porta RS-232C su una CPU della serie CS/CJ, una Scheda di comunicazione seriale (solo serie CS) o un Modulo di comunicazione seriale (porta RS-232C di CS1H/G-CPU□□, porte di CS1W-SCU21, porte di CS1W-SCB21, porta 1 di CS1W-SCB41 o porta 2 di CJ1W-SCU41), è necessario prendere in considerazione le differenze illustrate di seguito.

Prodotti precedenti	Codice del modello	Modifiche necessarie per i prodotti della serie CS/CJ	
		Cablaggio	Varie
Moduli Host Link della serie C	3G2A5-LK201-E C500-LK203 3G2A6-LK201-E	Il connettore a 25 pin è stato sostituito con un connettore 9 pin. I prodotti della serie CS/CJ non supportano i segnali ST1, ST2 e RT e non è pertanto necessario cablare i terminali corrispondenti.	**Le modifiche riportate di seguito sono necessarie per i sistemi sincronizzati con i segnali ST1, ST2 e RT.** Non è più possibile eseguire trasferimenti sincronizzati. È possibile eseguire trasmissioni full-duplex con i prodotti della serie CS/CJ, ma sarà necessario modificare l'hardware o il programma di comunicazione del computer host o entrambi. **Le modifiche riportate di seguito sono necessarie per i sistemi non sincronizzati con i segnali ST1, ST2 e RT.** È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di stringhe di testo di lunghezza differente all'interno dei frame o di specifiche differenti relative ai comandi della serie CS/CJ (vedere nota).
	C200H-LK201	Il connettore a 25 pin è stato sostituito con un connettore 9 pin.	È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di stringhe di testo di lunghezza differente all'interno dei frame o di specifiche differenti relative ai comandi della serie CS/CJ (vedere nota).
CPU della serie C	SRM1 CPM1 CPM1A CQM1-CPU□□-E C200HS-CPU□□-E C200HX/HG/HE-CPU□□-E C200HW-COM□□-E	Non sono state apportate modifiche a livello di cablaggio.	È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di specifiche differenti relative ai comandi della serie CS/CJ.

Prodotti precedenti	Codice del modello	Modifiche necessarie per i prodotti della serie CS/CJ	
		Cablaggio	Varie
CPU CVM1 o della serie CV	CVM1/CV-CPU□□-E	Non sono state apportate modifiche a livello di cablaggio.	È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di specifiche differenti relative ai comandi della serie CS/CJ.
Moduli Host Link CVM1 o della serie CV	CV500-LK201	Porta 1: il connettore a 25 pin è stato sostituito con un connettore 9 pin. Porta 2 impostata per RS-232C: il segnale SG associato in precedenza al pin 7 è ora associato al pin 9.	**Le modifiche riportate di seguito sono necessarie per le trasmissioni half-duplex che utilizzano CD.** Verificare l'eventuale presenza di problemi di ritardo quando si utilizzano le istruzioni SEND, RECV o CMND per avviare la comunicazione dal PLC o quando si inviano comandi dal computer host. Se necessario, passare alla trasmissione full-duplex. **Le modifiche riportate di seguito sono necessarie per le trasmissioni full-duplex che non utilizzano CD.** È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di specifiche differenti relative ai comandi della serie CS/CJ.

Nota Quando si utilizzano i comandi in modalità C, il numero di canali che è possibile leggere e scrivere per ogni frame, ovvero la lunghezza del testo, è differente per i Moduli Host Link della serie C e le schede o i Moduli di comunicazione seriale della serie CS/CJ. È possibile che il programma del computer host precedentemente utilizzato per i Moduli Host Link della serie C non funzioni correttamente per i PLC della serie CS/CJ. Verificare il programma del computer host prima di utilizzarlo e apportare le correzioni eventualmente necessarie per gestire stringhe di testo di lunghezza differente all'interno dei frame. Per ulteriori informazioni, fare riferimento al manuale *CS/CJ-series Communications Commands Reference Manual* (W342).

Porte RS-422A/485

Quando si passa da un sistema Host Link esistente a un altro che utilizza una porta RS-422A/485 su un Modulo di comunicazione seriale della serie CS (porta 2 di CS1W-SCB41) o un Modulo di comunicazione seriale della serie CJ (porta 1 di CJ1W-SCU41), è necessario prendere in considerazione le differenze illustrate di seguito.

Prodotti precedenti	Codice del modello	Modifiche necessarie per i prodotti della serie CS/CJ	
		Cablaggio	**Varie**
Moduli Host Link della serie C	3G2A5-LK201-E C200H-LK202 3G2A6-LK202-E	I pin di cablaggio sono stati modificati come descritto di seguito. SDA: da pin 9 a pin 1 SDB: da pin 5 a pin 2 RDA: da pin 6 a pin 6 RDB: da pin 1 a pin 8 SG: da pin 3 a non collegato FG: da pin 7 a guscio del connettore	È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di stringhe di testo di lunghezza differente all'interno dei frame o di specifiche differenti relative ai comandi della serie CS/CJ (vedere nota).
Scheda di comunicazione C200HX/HG/HE	C200HW-COM□□-E	Non sono state apportate modifiche a livello di cablaggio.	È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di specifiche differenti relative ai comandi della serie CS/CJ.
CPU CVM1 o della serie CV	CVM1/CV-CPU□□-E	Non sono state apportate modifiche a livello di cablaggio.	È possibile utilizzare i programmi del computer host senza apportare alcuna modifica purché vengano utilizzate le stesse impostazioni di comunicazione, ad esempio la velocità di trasmissione. Potrebbe tuttavia essere necessario modificare i programmi per consentire l'utilizzo di specifiche differenti relative ai comandi della serie CS/CJ.
Moduli Host Link CVM1 o della serie CV	CV500-LK201		

Nota Quando si utilizzano i comandi in modalità C, il numero di canali che è possibile leggere e scrivere per ogni frame, ovvero la lunghezza del testo, è differente per i Moduli Host Link della serie C e le schede o i Moduli di comunicazione seriale della serie CS/CJ. È possibile che il programma del computer host precedentemente utilizzato per i Moduli Host Link della serie C non funzioni correttamente per i PLC della serie CS/CJ. Verificare il programma del computer host prima di utilizzarlo e apportare le correzioni eventualmente necessarie per gestire stringhe di testo di lunghezza differente all'interno dei frame. Per ulteriori informazioni, fare riferimento al manuale *CS/CJ-series Communications Commands Reference Manual* (W342).

Index

T

U

Z

Storico delle revisioni

Il suffisso al numero di catalogo stampato sulla copertina del manuale indica il codice di revisione del documento.

Nella seguente tabella sono indicate le modifiche apportate al manuale nel corso di ciascuna revisione. I numeri di pagina si riferiscono alla versione precedente.

Codice di revisione	Data	Contenuto modificato
01	Aprile 2001	Stesura originale
02	Ottobre 2001	Aggiunta di informazioni relative alle CPU ad alta velocità della serie CS e della serie CJ (CS1G/H-CPU□□H e CJ1G/H-CPU□□H) in più punti del manuale.
03	Luglio 2002	Aggiunta di informazioni relative alle CPU CJ1M in più punti del manuale. Sostituzione dell'abbreviazione "PC" con "PLC" ad indicare il controllore programmabile. Ulteriori modifiche: **Pagine xvi e xviii:** sono state aggiunte nuove precauzioni. **Pagina xix:** l'elemento 2 in fondo alla pagina è stato modificato. **Pagina 28:** la descrizione della stringa di testo è stata modificata. **Pagina 167:** l'esempio di programmazione è stato modificato. **Pagine 168, 169, 265 e 266:** sono state aggiunte informazioni sugli alimentatori c.c. **Pagina 179:** sono state aggiunte precauzioni per l'utilizzo delle schedine di memoria. **Pagina 229:** l'illustrazione è stata modificata. **Pagina 262:** sono state aggiunte informazioni sul metodo di aggiornamento per temporizzatori e contatori. **Pagina 273:** sono state aggiunte precauzioni relative a DeviceNet. **Pagina 301:** le unità di misura relative alle velocità di elaborazione sono state corrette. **Pagina 304:** il valore del tempo di risposta degli interrupt è stato corretto. **Pagina 320:** le informazioni relative al supporto CJ1 per le istruzioni IOSP/IORS sono state modificate.
04	Settembre 2002	Aggiunta di informazioni relative alle CPU CJ1D in più punti del manuale. Ulteriori modifiche: **Pagina xv:** sono state aggiunte precauzioni relative alla funzione di backup. **Pagina xvi:** il testo della prima precauzione è stato riscritto e la precauzione relativa alla modalità operativa impostata all'avvio per i vari tipi di CPU è stata corretta. **Pagina xviii:** è stata aggiunta una precauzione relativa alla porta RS-232C (metà pagina). **Pagina 6:** sono state aggiunte informazioni sulle versioni di CX-Programmer. **Pagina 184:** le informazioni relative alla schedina di memoria riportate nella tabella sono state corrette. **Pagina 274:** l'intervallo di valori per l'impostazione del numero massimo di moduli è stato modificato (inizio pagina). **Pagina 294:** la nota è stata riscritta e modificata in un messaggio di Attenzione. **Pagina 303:** la nota 3 è stata riscritta. **Pagina 304:** sono state aggiunte nuove informazioni a metà tabella.
05	Aprile 2003	**Pagina 44:** Concetti di base sulla programmazione ladder riscritti. **Pagina 45:** Seconda informazione sulle restrizioni modificata. **Pagina 46:** Quarta informazione sulle restrizioni modificata. Sesta informazione sulle restrizioni rimossa. **Pagina 54:** Informazioni sui rung per cui è necessaria particolare attenzione o la riscrittura modificate. **Pagine 157 e 158:** Modifiche apportate alla tabella delle differenze tra task ciclici supplementari e normali. **Pagina 220:** Aggiunta la nota 5. **Pagina 226:** Informazioni sui moduli supportati modificate. **Pagina 233:** Informazioni relative alle cadute di tensione durante l'accesso alla memoria file modificate. **Pagina 264:** Nota aggiunta. **Pagina 273:** Nota aggiunta. **Pagina 276:** Dati delle tabelle modificati. Nota aggiunta sotto la prima tabella e nota modificata sotto la seconda tabella. **Pagina 310:** Informazioni sotto la prima tabella relative a CX-Programmer modificate. **Pagina 314:** Informazioni aggiunte alla seconda nota. **Pagina 326:** Informazioni aggiunte al passaggio 1 della procedura di registrazione dei dati.

Codice di revisione	Data	Contenuto modificato
06	Dicembre 2003	Informazioni sulle nuove funzioni supportate dalle nuove versioni delle CPU aggiunte (troppo numerose da elencare separatamente) . **Pagine da xi a xx:** Aggiornate le informazioni sulle precauzioni **Pagina 72:** Aggiunte note all'inizio della tabella e le istruzioni AND NOT e OR NOT. **Pagine 160, 201, 202, 228, 293 e 320:** Note aggiunte. **Pagina 189:** Informazioni su CX-Programmer versione 1.0 o successiva rimosse. **Pagina 191:** Informazioni aggiunte sopra la tabella e nella tabella, con aggiunta di una nuova tabella. **Pagine 197, 199, da 216 a 218, 230 e 231:** Informazioni per la sostituzione senza file dell'area dei parametri aggiunte. **Pagina 226:** Tabella espansa. **Pagina 294:** Nuova sezione 6-6-8 aggiunta. **Pagine da 337 a 339, 342, 344, 345 e 347:** Righe aggiunte alla tabella. **Pagina 339:** Informazioni aggiunte per BCMP2.
07	Luglio 2004	Modifiche nell'intero manuale per includere informazioni relative alle nuove funzioni supportate dall'aggiornamento dalla versione 2.0 alla versione 3.0 delle CPU della serie CS/CJ, incluse quelle elencate di seguito. **Pagina 8:** grafico modificato. **Pagina 32:** modifiche nelle colonne dei valori decimali ed esadecimali per il formato binario con segno. **Pagina 137:** aggiunte informazioni per descrivere le istruzioni TXD(236) ed RXD(235). Le informazioni su queste nuove istruzioni sono state aggiunte alle sezioni *3-23* e *3-24*. **Pagina 190:** informazioni modificate e aggiunte nella panoramica e nel grafico. Informazioni modificate e aggiunte nella tabella. **Pagine 192 e 193:** informazioni riorganizzate e modificate nella sezione *5-1-2*. **Pagina 194:** tabella espansa. **Pagina 198:** tabella espansa, note aggiunte e rimosse e informazioni sui file di sistema per CX-Programmer aggiunte. **Pagina 202:** modificato il formato del contenuto della cella da “numerico o testo” in “testo”. **Pagina 204:** tabella modificata e nota aggiunta. **Pagina 207:** aggiunte informazioni sui file delle tabelle dei simboli e sui file di commento. **Pagina 228:** aggiunte nuove informazioni in tabelle relative a verifica e stato di allocazione degli I/O. Aggiunte inoltre informazioni sulla funzione di backup semplice. **Pagina 229:** aggiunte informazioni sul confronto dei dati. **Pagine 232 e 233:** intestazioni di tabella corrette sostituendo CPU a memoria di I/O. **Pagina 233:** aggiunta nuova tabella. **Pagina 240:** aggiunte informazioni relative a verifica e stato di allocazione degli I/O. **Pagina 277:** aggiunte in tabella informazioni relative al Gateway seriale. **Pagina 285:** aggiunta nuova sezione *6-3-4,* sul Gateway seriale. **Pagine 345 e 364:** aggiunte informazioni alla tabella.